The Number Field Sieve integer factorization algorithm

Décio Luiz Gazzoni Filho
decio@decpp.net

Marco Antonio Torrez Rojas
matrojas@ime.usp.br

WCAP'05
September 28, 2005

Outline

- Introduction
  - Importance of factoring
  - Some factoring algorithms
  - Factoring by congruence of squares
  - Sieves
  - Strategy of NFS
- Overview of algebraic number theory
- Special NFS
- General NFS
- (Time permitting) Improvements and variants

Introduction

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Importance of factoring

> The problem of distinguishing prime numbers from composites, and of resolving composite numbers into their prime factors, is one of the most important and useful in all arithmetic.
>
> C. F. Gauss, Disquisitiones Arithmeticae (1801)

- Cryptographic applications
- Groups/fields: algebraic structure linked to factorization of order
- Number theory: Chinese Remainder Theorem, primality proving, factoring (!)

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Some factoring algorithms

- Notation: n composite, p prime factor of n
- Trial division: divide by $2, 3, 5, 7, 11, \ldots, p$; $\#$ divisions $\approx p/\log p$
- Pollard rho: perform random walk mod n and detect cycle mod p; cycle length $\sqrt{p}$ due to birthday paradox
- Pollard $p-1$: $a^{p-1} \equiv 1 \pmod{p}$, hence $\gcd(a^{p-1}-1, n) = p$; try exponents of the form $2^{50}3^{32}5^{22}7^{18}\cdots$; can use other grups (ECM)
- Congruence of squares (CFRAC, MPQS, NFS)

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares

- Squares in $(\mathbb{Z}/n\mathbb{Z})^*$ (quadratic residues)
- If n has k prime factors then $\sqrt{a} \pmod{n}$ has 2^k solutions
- Suppose $x^2 \equiv y^2 \pmod{n}$ but $x \not\equiv \pm y \pmod{n}$
- Then $n \mid (x+y)(x-y)$ but $n \nmid x \pm y$ (factors of n factors split between $x+y$ and $x-y$)
- So: $\gcd(x \pm y, n)$ is nontrivial factor of n
- How to construct congruences of squares?

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares (cont.)

- Let $x > \sqrt{n}$, so $x^2 \neq x^2 \bmod n$
- Find set X such that $\prod_{x \in X}(x^2 \bmod n)$ is a square
- This is a congruence of squares:

$$\prod_{x \in X} x^2 \equiv \prod_{x \in X} (x^2 \bmod n) \pmod{n}$$

- How to find set X?

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares (cont.)

- Try $\lceil\sqrt{n}\rceil, \lceil\sqrt{n}+1\rceil, \ldots$ Example ($n = 2041$):

$$
\begin{aligned}
46^2 \bmod 2041 &= 75 = 3 \times 5^2, \\
47^2 \bmod 2041 &= 168 = 2^3 \times 3 \times 7, \\
48^2 \bmod 2041 &= 263 \text{ is prime}, \\
49^2 \bmod 2041 &= 360 = 2^3 \times 3^2 \times 5, \\
50^2 \bmod 2041 &= 459 = 3^3 \times 17, \\
51^2 \bmod 2041 &= 560 = 2^4 \times 5 \times 7, \ldots
\end{aligned}
$$

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares (cont.)

- Let $X = \{46, 47, 49, 51\}$, so

$$46^2 \times 47^2 \times 49^2 \times 51^2 \equiv 2^{10} \times 3^4 \times 5^4 \times 7^2 \pmod{n},$$

yet

$$x = 46 \times 47 \times 49 \times 51 \not\equiv 2^5 \times 3^2 \times 5^2 \times 7 = y \pmod{n}.$$

- Thus $\gcd(x + y, n) = 157$ and $\gcd(x - y, n) = 13$, so $2041 = 13 \times 157$.
- Highly heuristic procedure
- How to build an algorithm out of this?

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares (cont.)

- Build *factor base* of small primes, discard values with factors not in factor base
- *Smooth* values are rare; avoid small factor bases
- Write factorization as vector of exponents: $(e_1, e_2, \ldots, e_k) = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$
- Squares have $(e_1, e_2, \ldots, e_k) \equiv (0, 0, \ldots, 0) \pmod 2$
- Build matrix of exponent vectors mod 2 and find vector in null space (i.e. linear dependence)
- LA algorithms formalize, speed up search for squares
- LA thm: linear dependence exists if $\#$ vectors $> k$
- Build exponent vectors from continued fractions (CFRAC) or by sieving quadratic polynomials (QS)
- Dense methods don't meet complexity bound; must use fast sparse methods (block Lanczos/Wiedemann)

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Factoring by congruence of squares (cont.)

- Matrix for example factorization
- Factor base: 2, 3, 5, 7
- Relations:

$$
\begin{aligned}
46^2 \bmod 2041 &= 75 = 3 \times 5^2, \\
47^2 \bmod 2041 &= 168 = 2^3 \times 3 \times 7, \\
49^2 \bmod 2041 &= 360 = 2^3 \times 3^2 \times 5, \\
51^2 \bmod 2041 &= 560 = 2^4 \times 5 \times 7
\end{aligned}
$$

- Matrix:

$$
\begin{bmatrix} 0 & 1 & 2 & 0 \\ 3 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \\ 4 & 0 & 1 & 1 \end{bmatrix} \equiv \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}
$$

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Sieves

- Number-theoretical algorithms on regularly spaced blocks of integers
- Cost of sieve amortized among integers in block
- Idea: consider $\mathbb{N} = 1, 2, 3, \ldots$ Multiples of 2 spaced by 2, multiples of 3 spaced by 3, etc.
- Costs to find factors $< p$ for integers $n+1, \ldots, n+j$:
  - Trial division: cost of division by each prime $< p$ (about $p/\log p$ divisions), times j
  - Sieving: for each prime $x < p$, find first multiple of x in list (cost: division by x) then mark every x-th element of list
  - Cost of sieve per element is $O(\log\log p)$. Sketch:

$$\sum_{\substack{x<p \\ x \text{ prime}}} \frac{1}{x} = \log\log p.$$

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Sieving polynomials

- Let $f(x) = a_d x^d + \ldots + a_1 x + a_0$ and $x_1, \ldots, x_k$ roots of $f(x)$ mod p (i.e. $f(x_i) \equiv 0 \pmod{p}$)
- Theorem: $f(x) \equiv f(x+p) \pmod{p}$
- Arithmetic progressions of multiples of p

$$f(x_1), f(x_1+p), f(x_1+2p), \ldots$$
$$\vdots$$
$$f(x_k), f(x_k+p), f(x_k+2p), \ldots$$

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Sieving to recognize smooth values

- Initialize $v_n = n, v_{n+1} = n + 1, \ldots$
- 'Mark' entries by dividing by current prime
- At end of sieve, entries with $v_i = 1$ are p-smooth
- Multiprecision operations too costly, so use approximate logarithms
- Can trade off sieve accuracy for performance
- Fact: if i not p-smooth then $v_i > p$
- Use crude approximations for logarithms since error margin is $\log p$ (most QS/NFS implementations use 8-bit integers)
- Sieving with small primes (< 30) costly yet doesn't influence much: usually not done, add 'fudge factor'
- Must trial divide reports to assert smoothness

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Strategy of NFS

- $\mathbb{Z}[\alpha_1], \mathbb{Z}[\alpha_2]$ algebraic number rings, ϕ_1, ϕ_2 homomorphisms $\phi_i : \mathbb{Z}[\alpha_i] \to \mathbb{Z}/n\mathbb{Z}$
- Find elements $\beta_i^2 \in \mathbb{Z}[\alpha_i]$ such that β_i^2 is a square and $\phi_1(\beta_1) \equiv \phi_2(\beta_2) \pmod{n}$
- Compute square roots β_i in number ring, hope for $\phi_1(\beta_1) \not\equiv \pm\phi_2(\beta_2) \pmod{n}$
- $\gcd(\phi_1(\beta_1) \pm \phi_2(\beta_2), n)$ non-trivial factors of n
- Use techniques of previous slides (factor bases, etc.)
- Main feature: ϕ_i 'look like' polynomials; can use sieving techniques

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Why is NFS fast?

- Let $L_n[u, v] = \exp\left((v + o(1))(\log n)^u (\log\log n)^{1-u}\right)$
- MPQS cost: $L_n[1/2, 1]$; GNFS cost: $L_n[1/3, 1.923]$
- Source of difference?
- Performance of congruence-of-squares algorithms depends on size of integers examined for smoothness
- If relations behave like $x(n)$ then cost is $L_x[1/2, \sqrt{2}]$
- MPQS: $x(n) = n^{1/2}$ (so half the digits of n)
- If $x(n) = n^{1/k}$ for fixed k, cost is $L_n[1/2, \sqrt{2/k}]$
- NFS: $x(n, d, M) = 2dn^{2/d}M^{d+1}$, d degree of algebraic extension and M size of sieving region
- For optimal choice of d, M, $x(n) = n^{o(1)}$ as $n \to \infty$; is asymptotically better than any fixed k, hence improved running time

Introduction
Importance of factoring
Some factoring algorithms
Factoring by congruence of squares
Sieves
Strategy of NFS
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Overview of algebraic number theory

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

What is a number field?

- Fix irreducible polynomial $f(x) = a_d x^d + \ldots + a_0$, $a_i \in \mathbb{Z}$ and α an irrational ($\mathbb{R}$ or $\mathbb{C}$) root of $f(x)$
- Extend $\mathbb{Q}$ by adjoining α: $\mathbb{Q}(\alpha)$ is a *number field*
- Add/multiply as polynomials in α and reduce using

$$\alpha^d = -\frac{a_{d-1}\alpha^{d-1} + \ldots + a_0}{a_d}$$

- Elements of the form $\beta = c_0 + \ldots + c_{d-1}\alpha^{d-1}$, $c_i \in \mathbb{Q}$
- For each β there is a unique irreducible least-degree $f(x)$ with root β (the *minimal polynomial*)
- Canonical example: $f(x) = x^2 + 1$, $\alpha = i = \sqrt{-1}$ (Gaussian integers)
- Also $-i$, but conjugates are indistinguishable
- *Algebraic number theory*: study of arithmetical properties of algebraic number fields

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

What is a number field? (cont.)

- How to define analogue of integers in a number field?
- Must have familiar properties ($\beta \in \mathbb{Q}(\alpha)$)
  1. Must form a ring (as $\mathbb{Z}$ is subring of $\mathbb{Q}$)
  2. $\beta \in \mathbb{Q} \Rightarrow \beta \in \mathbb{Z}$
  3. $n\beta$ algebraic integer for some $n \in \mathbb{Z}$
- Ring of integers: elements with *monic* minimal polynomials (leading coefficient 1)
- $\mathbb{Z}[\alpha] = c_0 + \ldots + c_{d-1}\alpha^{d-1}$, $c_i \in \mathbb{Z}$ subring of full ring of integers
- Example: $\phi = (1+\sqrt{5})/2$ root of $f(x) = x^2 - x - 1$; $\mathbb{Z}[\phi] \supset \mathbb{Z}[\sqrt{5}]$
- $\mathbb{Z}[\alpha]$ may be full ring of integers (e.g. $\mathbb{Z}[i]$)

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Some properties of rational integers

- Division algorithm ($a = qb + r, 0 \leq r < b$)
- Existence of GCDs
- Units (invertible elements): only ± 1
- Existence, uniqueness of factorization into primes

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Gaussian integers

- $\mathbb{Z}[i] = \{a + bi, a, b \in \mathbb{Z}, i = \sqrt{-1}\}$
- Similar properties to rational integers
- **Norm**: $N(a + bi) = a^2 + b^2$
- Division algorithm: $a = bq + r$, $0 \leq N(r) < N(b)$
- Units: $\pm 1, \pm i$
- Unique Factorization Domain (UFD)

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

‘Badly behaved’ number fields

- No division algorithm (can't ensure $r < N(b)$)
- Hence, no GCDs
- If division algorithm exists: Euclidean domain
- Units: infinite unless imaginary quadratic field
- Example ($\mathbb{Z}[\sqrt{2}]$): $(1+\sqrt{2})^n, n \in \mathbb{Z}$
- $\frac{1}{1+\sqrt{2}} \frac{1-\sqrt{2}}{1-\sqrt{2}} = \frac{1-\sqrt{2}}{1^2-\sqrt{2}^2} = 1-\sqrt{2}$
- Theorem (Dirichlet): group of units is abelian, finitely generated of rank $r_{\mathbb{R}} + r_{\mathbb{C}}/2 - 1$
- Existence of primes and factorization
- However: may not be unique
- Consider $\mathbb{Z}[(1+\sqrt{-5})/2]$ and primes $3, 7, 1 \pm 2\sqrt{-5}$
- $21 = 3 \cdot 7 = (1+2\sqrt{-5})(1-2\sqrt{-5})$
- Unique factorization restored by use of **ideals**

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
‘Badly behaved’ number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Definitions

- Number field $K = Q(\alpha)$, number ring $\mathbb{Z}_K$, algebraic number $\beta = b_0 + b_1\alpha + \ldots + b_{d-1}\alpha^{d-1}$
- **Minimal polynomial**: unique irreducible least-degree polynomial $f(x)$ with $\mathbb{Z}$ coefficients and $f(\alpha) = 0$
- $\alpha_1, \ldots, \alpha_d$ all roots of $f(x)$, $\alpha = \alpha_1$ (arbitrarily)
- **Degree** of K: d
- **Index** of α: $[\mathbb{Z}_K : \mathbb{Z}[\alpha]]$
- **Conjugates** of β: $b_0 + b_1\alpha_i + \ldots + b_{d-1}\alpha_i^{d-1}$
- **Trace** of β ($\mathrm{Tr}(\beta)$): sum of all conjugates
- **Norm** of β ($\mathcal{N}(\beta)$): product of all conjugates
- Is a multiplicative function
- Is a rational fraction (integer if β algebraic integer)
- Example: i root of $x^2 + 1$, so $-i$ also root; norm is $\mathcal{N}(a + bi) = \mathcal{N}(a - bi) = (a + bi)(a - bi) = a^2 + b^2$
- **Basis**: d-tuple of elements that generate K
- **Integral basis**: d-tuple of elements that generate $\mathbb{Z}_K$

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Definitions (cont.)

- **Discriminant** of set of algebraic numbers $\gamma_1, \ldots, \gamma_d$: determinant of $\mathrm{Tr}(\gamma_i \gamma_j)$
- Discriminant of an integral basis independent of choice of basis: **discriminant of** K
- Discriminant of $f(x)$: square times discriminant of K
- **Ideal**: J ideal if closed under addition and, for $a \in K, b \in J$, $ab \in J$
- In particular, is subring of K
- Informally: set of multiples of some element(s)
- Example: ideals of $\mathbb{Z}$: $n\mathbb{Z} = \{0, \pm n, \pm 2n, \ldots\}$
- Gives rise to ring K/J: an element α of K is congruent to β if $\alpha - \beta \in J$
- **Norm of ideal**: cardinality of K/J
- **Principal ideal** (α): ideal generated by α only
- **Principal Ideal Domain (PID)**: all ideals principal
- Theorem: PID $\Leftrightarrow$ UFD

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Digression on ideals

- Again consider $\mathbb{Z}_K = \mathbb{Z}[(1+\sqrt{-5})/2]$ and factorizations $21 = 3 \cdot 7 = (1+2\sqrt{-5})(1-2\sqrt{-5})$
- Let $\alpha = 1+2\sqrt{-5}, \lambda = 2+\sqrt{-5}$
- Then $\alpha^2/\lambda = -2+3\sqrt{-5}, 3^2/\lambda = 2-\sqrt{-5}$ are integers of $\mathbb{Z}_K$
- If $p(x) = 0$ for $x = \lambda$, then $p(x^2) = 0$ for $x = \sqrt{\lambda}$, so λ is integer in some other number field K'
- Hence $\alpha/\sqrt{\lambda}, 3/\sqrt{\lambda}$ also integers in K', so $\sqrt{\lambda}$ is common divisor of $\alpha, 3$ in K'
- In fact, $\sqrt{\lambda} = \alpha(-2\alpha/\sqrt{\lambda}) - 3((12-3\sqrt{-5})/\sqrt{\lambda})$, so $\sqrt{\lambda}$ is GCD of $\alpha, 3$ in K'
- Similarly, $\sqrt{\kappa}$ GCD of $7, 1-2\sqrt{-5}$ for $\kappa = 2+3\sqrt{-5}$
- In K', $21 = \sqrt{\lambda}\sqrt{\overline{\lambda}}\sqrt{-\kappa}\sqrt{-\overline{\kappa}}$
- Factorizations in K from pairings of factors in K'
- Non-principal ideals: multiples of element not in K

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Arithmetic of ideals

- Addition: $I + J = \{x + y, x \in I, y \in J\}$
- Informally: GCD
- Example: $6\mathbb{Z} + 10\mathbb{Z} = 2\mathbb{Z}$ since $n(2 \cdot 6 - 10) = 2n$
- Multiplication: $IJ = \{\sum_i x_i y_i, x \in I, y \in J\}$
- $IJ \subseteq I \cap J$, and if I, J coprime $= IJ = I \cap J$
- $6\mathbb{Z} \cdot 10\mathbb{Z} = 60\mathbb{Z}$ since $6m \cdot 10n = 60mn$, while $6\mathbb{Z} \cap 10\mathbb{Z} = 30\mathbb{Z}$
- Divisibility (in $\mathbb{Z}_K$): $I \mid J$ if $J \subset I$
- I is **prime ideal** of K if, for $a, b \in K$, $ab \in I$ implies $a \in I$ or $b \in I$ (similar property for prime numbers: if $p \mid ab$ then $p \mid a$ or $p \mid b$)
- Prime ideals of $\mathbb{Z}$: $p\mathbb{Z}$ for p prime
- **Unique factorization into prime ideals** in NFs
- Example: $30\mathbb{Z} = 2\mathbb{Z} \cdot 3\mathbb{Z} \cdot 5\mathbb{Z}$

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Decomposition of prime numbers

- $p\mathbb{Z}_{\mathbb{K}} = \prod_{i=1}^{g} \mathfrak{p}_i^{e_i}$
- e_i: ramification index
- $f_i = [\mathbb{Z}_k/\mathfrak{p}_i : \mathbb{Z}/p\mathbb{Z}]$: degree
- $\sum_{i=1}^{g} e_i f_i = \deg(K)$
- If $g = 1, e_1 = 1$, then p is **inert** (i.e. $p\mathbb{Z}_K = p\mathbb{Z}$)
- If $g = \deg(K)$, then p **splits completely**
- If $e_i \geq 2$ for some i, then p **ramifies**
- Ramified primes those that divide discriminant of K
- Algorithm for case $p \nmid f$ (f index of α):
  - Minimal poly. $T(x) \equiv \prod_{i=1}^{g} T_i(x)^{e_i} \pmod{p}$
  - Then $\mathfrak{p}_i = (p, T_i(\alpha)) = p\mathbb{Z}_K + T_i(\alpha)\mathbb{Z}_K$
  - $f_i = \deg(T_i)$

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Ideal classes

- Consider ideals modulo principal ideals
- Equivalence relation: $I \sim J$ if there exists $\alpha, \beta \in K$ such that $\alpha I = \beta J$
- Gives rise to classes of equivalent ideals, which form the class group $\mathrm{Cl}(K)$
- Class number $h = \#\mathrm{Cl}(K)$ is finite, measures extent of failure of unique factorization
- All principal ideals equivalent, so $h = 1$ in a PID
- Non-principal ideals: multiples of element in extension of K of degree at most h

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Computational algebraic number theory

- Tasks of computational algebraic number theory
  - Compute integral basis of $\mathbb{Z}_K$
  - Find decomposition of prime numbers
  - Compute a system of fundamental units
  - Compute the class number and class group
  - Determine if an ideal $I \in \mathbb{Z}_K$ is principal; if it is, find a generator $\alpha \in K$ such that $i = \alpha\mathbb{Z}_K$
- Efficient algorithms exist for *simple* fields (small discriminant, class number, etc.)
- Special NFS: simple fields
- General NFS: complicated number field, must avoid many tasks above

Introduction
Overview of algebraic number theory
What is a number field?
Some properties of rational integers
Gaussian integers
'Badly behaved' number fields
Definitions
Digression on ideals
Arithmetic of ideals
Decomposition of prime numbers
Ideal classes
Computational algebraic number theory
Special NFS
General NFS
Complexity sketch

Special NFS

Introduction
Overview of algebraic number theory
Special NFS
Overview
Sieving in $\mathbb{Z}[\alpha]$
Final steps
General NFS
Complexity sketch

Overview

- Let $n = r^e + s$ with $r, |s|$ small
- Find polynomial and root mod n
- Degree chosen by complexity analysis (currently $d = 5, 6$)
- Example: $3^{239} - 1$
- $f(x) = x^5 - 3$, $m = 3^{48}$: $f(m) = 3n$
- Will work in number field $\mathbb{Z}[\alpha]$, with α complex root of $f(x)$
- Field is likely to be simple (UFD, etc.)
- Need second number field for NFS
- Hard to find other polynomial, so use $g(x) = x - m$
- Build factor base of prime ideals of $\mathbb{Z}[\alpha]$ and rational primes (for $g(x)$)
- Search for $a - b\alpha, a - bm$ simultaneously smooth
- Homomorphism: $\phi : \alpha \mapsto m \pmod{n}$

Introduction
Overview of algebraic number theory
Special NFS
Overview
Sieving in $\mathbb{Z}[\alpha]$
Final steps
General NFS
Complexity sketch

Sieving in $\mathbb{Z}[\alpha]$

- Norm encodes ideal factorization of $a - b\alpha$
- So: sieve norms
- Consider homogenous polynomial $F(x, y) = a_0 y^d + a_1 x y^{d-1} + \ldots + a_d x^d = y^d f(x/y)$
- Norm of $a - b\alpha$ is $F(a, b)$
- Fix b, sieve over a, change b, repeat (line sieving)
- Theorem: every ideal that contains $a - b\alpha$ is first degree ($f = [\mathbb{Z}_K/\mathfrak{p} : \mathbb{Z}/p\mathbb{Z}] = 1$)
- There could be multiple ideals of same norm
- Find linear factors of $f(x)$ mod p, associate each factor $x - c_i$ with an ideal of norm p
- Ideal is $(p, c_i - \alpha)$
- To find which ideal divides $a - b\alpha$: if $a - bc_i \equiv 0 \pmod{p}$, then divisor is $(p, c_i - \alpha)$

Introduction
Overview of algebraic number theory
Special NFS
Overview
Sieving in $\mathbb{Z}[\alpha]$
Final steps
General NFS
Complexity sketch

Sieving in $\mathbb{Z}[\alpha]$ (cont.)

- Example: α root of $f(x) = x^5 - 3$
- Norm of $a - b\alpha$ is $x^5 - 3y^5$
- Consider $7 - 4\alpha$: $\mathcal{N}(7 - 4\alpha) = 13735 = 5 \cdot 41 \cdot 67$
- $f(x)$ has linear factors:
  - $x - 3$ mod 5
  - $x - 11, x - 12, x - 28, x - 34, x - 38$ mod 41
  - $x - 52$ mod 67
- Solve $a - bc \equiv 0 \pmod{p_i}$ for c
  - $p_1 = 5$: $c = 5$
  - $p_2 = 41$: $c = 12$
  - $p_3 = 67$: $c = 52$
- Hence: $7 - 4\alpha = (5, 3 - \alpha)(41, 12 - \alpha)(67, 52 - \alpha)$
- As principal ideals:
  $(2 - \alpha + \alpha^3 - \alpha^4)(2 + \alpha + \alpha^2 + \alpha^3)(1 + \alpha^2 + \alpha^3)$

Introduction
Overview of algebraic number theory
Special NFS
Overview
Sieving in $\mathbb{Z}[\alpha]$
Final steps
General NFS
Complexity sketch

Final steps

- Suppose enough (a, b) pairs were found such that $a - b\alpha$ and $a - bm$ are simultaneously smooth
- Use linear algebra to find set $\mathcal{S}$ of relations with even exponents
- Then $\phi(\prod a - b\alpha) \equiv \prod(a - bm) \pmod{n}$ and both sides are squares, right?
- Problem: 4 is a square but -4 isn't
- Must find unit contribution (easy in simple fields)
- Square root: just divide exponents by 2
- How to write out the square root?
- Must find generators of prime ideals in the factor base (must exist as the number field is a PID)

Introduction
Overview of algebraic number theory
Special NFS
Overview
Sieving in $\mathbb{Z}[\alpha]$
Final steps
General NFS
Complexity sketch

General NFS

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Overview

- If n not of special form, still possible to find polynomial, but number field will no longer be simple
- Assume degree d; expand n in base $\lfloor m = n^{1/d} \rfloor$, so that $n = c_0 + c_1 m + \ldots + c_{d-1} m^{d-1} + m^d$
- Polynomial is $x^d + c_{d-1} x^{d-1} + \ldots + c_1 x + c_0$ and root m mod n
- Much bigger coefficients ($O(n^{1/d}$ instead of ϵ), so values sieved will be larger and density of smooth numbers will decrease
- Can use different values of m (some polynomials have better properties)
- No longer possible to find fundamental units, generators of prime ideals, etc.

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Obstructions

- Two main problems:
  - $\mathbb{Z}[\alpha]$ likely not the full ring of integers $\mathbb{Z}_K$
  - Most likely not a PID
- Let $\beta = \prod a - b\alpha$, such that exponents of prime ideals are all even
- May not be a square
- Four obstructions
  - β may not be the square of an ideal, since we use the primes of $\mathbb{Z}[\alpha]$, not $\mathbb{Z}_K$
  - Even if β is the square of an ideal, may not be a principal ideal
  - Unit contribution
  - Square root of β may be in $\mathbb{Z}_K \setminus \mathbb{Z}[\alpha]$
- Solution: quadratic characters

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Quadratic characters

- Adleman, 1991
- Probabilistic method to identify squares
- Squareness test using quadratic residues
- Example: $(\frac{7}{11})$ = -1, so 7 not a square
- $1, 4, 9, 16, 25, \ldots$ always quadratic residues
- As in a pseudoprimality test, if quadratic residue modulo sufficiently many primes, likely to be a square
- If quadratic residues are 'random', then chance of non-square being quadratic residue for a given prime is $1/2$
- For k tests, chance of failure is 2^{-k}
- Density of n-bit squares is $n/2$ bits, so a bit more than $n/2$ tests should suffice

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Quadratic characters (cont.)

- Can *construct* squares
- Example: $(\frac{3\cdot 5}{11}) = 1$, $(\frac{5\cdot 7}{11}) = -1$, $(\frac{3\cdot 7}{11}) = -1$, so $(\frac{3\cdot 5}{11})(\frac{5\cdot 7}{11})(\frac{3\cdot 7}{11}) = 1 \times -1 \times -1$ looks like (and indeed is) a square
- Append quadratic characters to relation vectors (with $1 \mapsto 0, -1 \mapsto 1$) and search for linear dependence (all characters 1)
- Squareness testing in number fields: $(\frac{a-bm}{q})$ for q not in the factor base
- Can estimate 'dimension' of obstructions (as a vector base over GF(2))
- Use more characters than dimension of obstructions
- Result is likely to be a square

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Square roots

- Must compute $s = \phi(\sqrt{\prod a - b\alpha})$
- Trivial in SNFS: if $s = \phi(\sqrt{\mathfrak{p}_1^{e_1} \cdots \mathfrak{p}_B^{e_B}})$, then $s = \phi(\mathfrak{p}_1^{e_1/2} \cdots \mathfrak{p}_B^{e_B/2}) = \phi({\mathfrak{p}_1}^{e_1/2}) \cdots \phi({\mathfrak{p}_{\mathfrak{B}}}^{e_B/2})$
- Since ϕ is computed mod n, no size explosion
- Unable to compute square root directly in GNFS
- Arithmetic with expanded s *extremely* expensive
- Must use fast multiplication methods (FFT, etc.)
- Initially: find minimal polynomial mod q (q not in factor base), Hensel lift up to a certain bound
- Couveignes: use CRT instead; 'explicit' CRT allows reduction of $(c_1 \bmod q_1, \ldots, c_k \bmod q_k)$ mod n
- Problem: two square roots modulo each prime; must find correct combination of signs
- If extension degree odd, $\mathcal{N}(-\alpha) = -\mathcal{N}(\alpha)$; compute norm of $\prod a - b\alpha$ and compare signs
- Heuristic method of Montgomery solved the problem

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

An implementation of GNFS

```
{nfs(n)=l=lo      g(n);d=(3 *l/log(l))^(1/3)\1;k=3*l \log(2);m=n^(1/d)\1;t=d;s=n
;H=vector(d+1     ,i,1);(p( i)=prime(i));for(i=1,d,s -=H[i]*m^t;H[i+1]=s\m^t--);
(f(x)=eval(Po     l(H)));(F (x,y)=y^d*f(x/y));(G(x,y )=x-m*y;);(h(x)=eval(deriv(
Pol(H))));t=B=    precprime (2*exp((l*log(l)^2)^(1/3 )));(z(i)=primepi(i));P=z(B
);R=vector(P,i    ,lift(pol rootsmod(f(x),p(i))));q= vector(k,i,while(!r=polroot
smod(f(x),t=nex   tprime(t+ 2)),);[t,l               ift(r[1])])
;K=exp(log(n)/2   00);(L(B) =log(B)/lo               g(K)\/1);w=
vectorsmall(P,i,  L(p(i))); b=r=0;A=ve              ctor(P);for
(i=2,P,A[i]=A[i-  1]+#R[i]) ;o=A[P]+#R               [P];s=o+P+k
+1;M=matrix(s,s); N=vector( s);while(1               ,b++;e=vect
orsmall(8*B+1,a,L (abs(F(a- 4*B,b)*G(a-4*B,b))+1));f or(i=1,P,t=p(i);for(j=1,#R[
i],forstep(k=-4*B+1+b*R[i][ j]%t,4*B,t,e[k+4*B]-=w[i ]));for(j=1,if(t<sqrt(B),lo
g(B)/log(t),1),forstep(k=-4 *B+1+(-m*b)%t^j,4*B,t^j, e[k+4*B]-=w[i])));for(t=1,8
*B+1,if(e[t]<L(B^2)&&gcd(a= t-4*B,b)==1,C=factorint( abs(F(a,b)));D=if(G(a,b),fa
ctorint(abs(G(a,b))),[]);if (C&&D&&C[#C~,1]<=B&&D[#D ~,1]<=B,if(r++==s+1,break(2
));N[r]=[a ,b];for(i=1,#C~, c=z(C[i,1]                                 );for(j=1,
#R[c],if(( a-b*R[c][j])%C[i ,1]==0,M[r                                 ,A[c]+j]=C
[i,2])));f  or(i=1,#D~,M[r, o+z(D[i,1]                                 )]=D[i,2])
;for(i=1,k  ,M[r,o+P+i]=kro necker(a-b                                 *q[i][2],q
[i][1])<0;   M[r,s]=G(a,b)< 0);))));S=               lift(matker(Mod(M~,2)));for
(i=1,#S,V=   M~*S[,i];v=lif t(prod(j=1               ,P,Mod(p(j),n)^(V[o+j]/2)))
;if(#(U=nf    factor(nfinit (f(y)),x^2               -h(y)^2*prod(j=1,#S~,Mod((N
[j][1]-N[j    ][2]*y)^S[j,i ],f(y)))))               ~>1,(u(y,x=0)=eval(lift(U[1
,1])));g=g     cd(u(m)-h(m) *v,n);if(g               >1&g<n,break)));return(g);}
```


Introduction
Overview of algebraic number theory
Special NFS
General NFS
Overview
Obstructions
Quadratic characters
Square roots
An implementation of GNFS
Complexity sketch

Complexity sketch

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Complexity sketch

- $L(X) = L_X[1/2, 1] = \exp(\sqrt{2 \log X \log \log X})$
- Theorem: sequence of integers $< X$ and B-smooth ($B = L(X)^{1/\sqrt{2}}$); after about $L(X)^{\sqrt{2}}$ elements, some subset of them is a square
- Need $F(a, b), G(a, b) = a - bm$ smooth: require $F(a, b)G(a, b)$ smooth
- Coefficients of $F(a, b)$ and m bounded by $n^{1/d}$
- Region of sieving is $|a|, |b| \leq M$
- $|F(a, b)G(a, b)|$ bounded by $2(d + 1)n^{2/d}M^{d+1}$
- Using theorem: $M^2 = L(X)^{\sqrt{2}}$
- We have

$$\log X \sim \log(2(d+1)) + \frac{2}{d} \log n + (d+1)\sqrt{\frac{1}{2} \log X \log \log X}$$

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Complexity sketch (cont.)

- Term $\log(2(d+1))$ too small, discarded

$$\log X \sim 2/d \log n + d\sqrt{1/2 \log X \log\log X}$$

- Find minimum (using derivatives)

$$\frac{X'}{X} = -2/d^2 \log n + \sqrt{1/2 \log X \log\log X} + \\ + \frac{dX'(1+\log\log X)}{4X\sqrt{1/2 \log X \log\log X}}$$

- Setting $X' = 0$:

$$d = (2\log n)^{1/2}(1/2 \log X \log\log X)^{-1/4}$$

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Complexity sketch (cont.)

- Replacing d in the original expression:

$$\log X = 2(2\log n)^{1/2}(1/2\log X \log\log X)^{1/4}$$
$$(\log X)^{3/4} = 2(2\log n)^{1/2}(1/2\log\log X)^{1/4}$$

- Taking logs again: $\frac{3}{4}\log\log X = \frac{1}{2}\log\log n$

$$(\log X)^{3/4} = 2(2\log n)^{1/2}(1/2\log\log X)^{1/4}$$
$$\log X = \frac{4}{3^{1/3}}(\log n)^{2/3}(\log\log n)^{1/3}$$

- Complexity of NFS is

$$L(X)^{\sqrt{2}} = \exp\left((64/9)^{1/3}(\log n)^{1/3}(\log\log n)^{1/3}\right)$$

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch

Improvements and variants

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

Large prime variation

- Initially consider only $a - bm$ values
- Allow *partial* relations: after removing factors up to p, have remainder $< p^2$ (or smaller bound)
- Fact: remainder is prime (the *large prime*)
- Dramatic increase in relations found
- Can't enlarge factor base
- Instead, pair up relations with same large prime:

$$p_1^{e_1} \cdots p_k^{e_k} P \times p_1^{e'_1} \cdots p_k^{e'_k} P = p_1^{e_1+e'_1} \cdots p_k^{e_k+e'_k} P^2$$

- Paired partial relations behave as full relations: if $\prod x_i$ is a square then so is $\prod x_i P_i^2$
- Large prime collisions easy to find after collecting many relations, due to birthday paradox

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

Double large prime variation

- Allow reports with two large primes (or a large prime and a large prime ideal)
- If two large primes, remainder in range $[p^2, p^3]$ may be prime or product of two primes
- Apply cheap compositeness test (say base-3 SPRP) and throw away if probable prime
- Pairing up no longer enough: $x_1 P_1 P_2 \times x_2 P_1 P_3 = x_1 x_2 P_1^2 P_2 P_3$. Instead, find *cycles* among relations using graph algorithms
- What about three large primes?
- Reports may have 1, 2 or 3 large prime factors: compositeness test can't tell apart 2nd and 3rd case
- NFSNET using 2 large primes and 2 large prime ideals

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

Fast linear algebra

- Must solve sparse linear systems over $\mathrm{GF}(2)$ for congruence-of-squares methods
- Dense methods wasteful, don't meet time bound
- Structured Gaussian elimination acceptable for small factorizations
- Later, sparse methods (Lanczos, conjugate gradient) modified for $\mathrm{GF}(2)$
- Problem: half of $\mathrm{GF}(2)$ vectors self-orthogonal
- Use extensions (LaMacchia and Odlyzko) or subspaces (Montgomery) of $\mathrm{GF}(2)$
- Blocking to take advantage of bitwise operations
- Also: Wiedemann's method
- Research problem: large-scale distributed algorithm

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

GNFS polynomial selection

- Brian Murphy's PhD thesis
- RSA-155: 'the yield (...) is approximately 13.5 times that of a skewed pair of average yield'
- Search time only 100 (out of 8400) MIPS-years
- Numerical optimization problem
- Main techniques:
  - Skewed polynomials ($|a^d| < |a^{d-1}| < \ldots < |a^0|$)
  - Force many roots mod small p
  - Leading coefficient product of small primes to add projective roots
  - Reduce size of relations (reduced coefficients, choice of sieving region, location of real roots)
- Research problem: find good degree-d ($d > 2$) polynomial pairs

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

Lattice sieving

- Split factor base in two, bounds B_0 and B_1
- Fix a prime $B_0 < q < B_1$ (the *special-q*)
- Compute lattice of (a, b) pairs such that $q \mid a - bm$
- Sieve points of this lattice, change q and repeat
- Pollard on F_9: 'An Infinitely Skilful Programmer can get 83% of the solutions for 8.6% of the work.' Why?
- LS relations are B_0-smooth with 1 factor in $[B_0, B_1]$
- Most B_1-smooth relations of this form
- Most candidate relations *not* of this form
- % integers sieved: $W = \sum 1/q$, % solutions lost: $L = \rho(\log(a + bm)/\log B_1)/\rho(\log(a + bm)/\log B_0)$

B_0/B_1	W	L
0.3	0.0855	0.167
0.2	0.1143	0.083
0.1	0.1636	0.022

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant

Coppersmith's variant

- Multiple polynomial variant of QS much faster
- How about multiple polynomial NFS?
- Problem: distinct ideals for each number field
- Must enlarge factor base
- Coppersmith: use k polynomials and bound B/k
- Sieve $a - bm$ then ECM each polynomial (sieving inefficient due to irregular spacing)
- Complexity lowered to $L_n[1/3, 1.902]$
- Pomerance: 'crossover point (...) in the thousands of digits'

Introduction
Overview of algebraic number theory
Special NFS
General NFS
Complexity sketch
Improvements and variants
Large prime variation
Double large prime variation
Fast linear algebra
GNFS polynomial selection
Lattice sieving
Coppersmith's variant