TECHNISCHE UNIVERSITÄT DARMSTADT

On Solving Number Theoretic Problems with Lattice Reduction

Alexander May

Department of Computer Science

TU Darmstadt

IPAM 2006 - Number Theory & Cryptography: Open Problems

Outline of the talk

TECHNISCHE UNIVERSITÄT DARMSTADT

- Solve polynomial equations
- Computing
  - e-th roots
  - RSA secret key
  - Factoring
  - Square roots mod p, mod N
- Coppersmith's method
- Automated computation of bounds
- Cyclic lattices & NTRU

A word about lattices

Let $v_1, \ldots, v_n \in \mathbb{Q}^n$ be linearly independent vectors.

$$L := \left\{ x \in \mathbb{Z}^n \mid x = \sum_{i=1}^{n} a_i v_i \text{ with } a_i \in \mathbb{Z}. \right\}$$

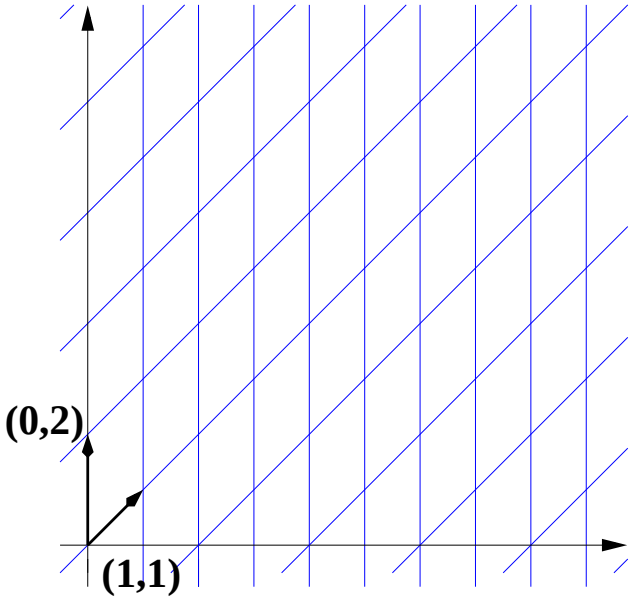


Fig.: Lattice spanned by $\{(0,2),(1,1)\}$

A word about lattices

Let $v_1, \ldots, v_n \in \mathbb{Q}^n$ be linearly independent vectors.

$$L := \left\{ x \in \mathbb{Z}^n \mid x = \sum_{i=1}^{n} a_i v_i \text{ with } a_i \in \mathbb{Z}. \right\}$$

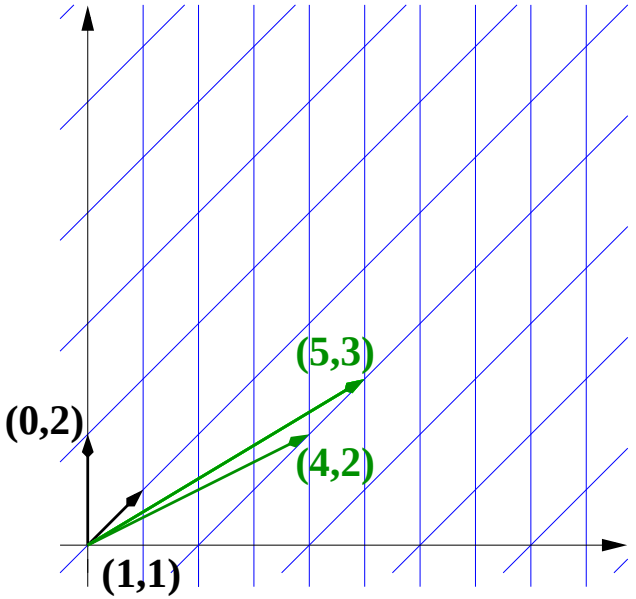


Fig.: Lattice spanned by $\{(0,2),(1,1)\}$

A word about lattices

Let $v_1, \ldots, v_n \in \mathbb{Q}^n$ be linearly independent vectors.

$$L := \left\{ x \in \mathbb{Z}^n \mid x = \sum_{i=1}^{n} a_i v_i \text{ with } a_i \in \mathbb{Z}. \right\}$$

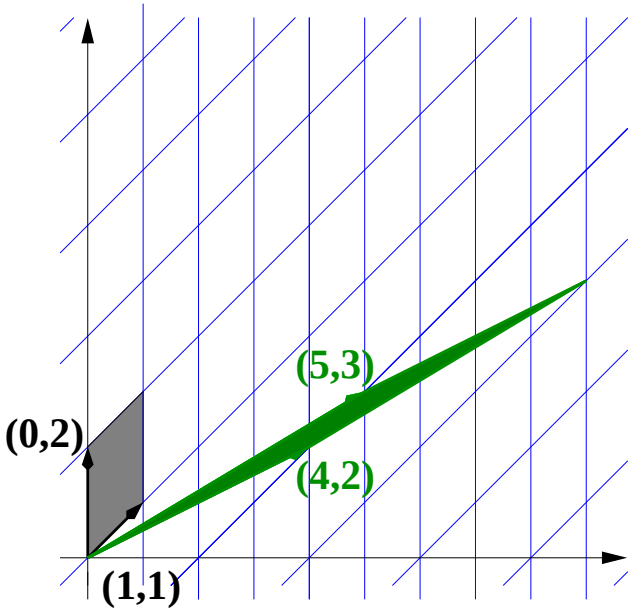


Fig.: Lattice spanned by $\{(0,2),(1,1)\}$

Lagrange- and L^3-algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem[Lagrange]: *Let L be a two-dimensional lattice. Then one can find in polynomial time a shortest vector $v \neq 0$ of L with*

$$\|v\| \leq \sqrt{2 \det(L)}.$$

Lagrange- and L^3-algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem[Lagrange]: *Let L be a two-dimensional lattice. Then one can find in polynomial time a shortest vector $v \neq 0$ of L with*

$$\|v\| \leq \sqrt{2 \det(L)}.$$

Theorem[Lenstra, Lenstra, Lovász]: *Let L be a lattice spanned by $v_1, \ldots, v_n$. Then one can find in polynomial time a basis $b_1, \ldots, b_n$ of L with*

$$\|b_1\| \leq 2^{\frac{n-1}{4}} \det(L)^{\frac{1}{n}}.$$

RSA Problem

Given: $N = pq$, $e \in \mathbb{Z}^*_{\phi(N)}$ and $c = m^e \bmod N$

Find: $m \in \mathbb{Z}_N$

- **Relaxation 1: Small e, m**
  Ptime if $m < N^{\frac{1}{e}}$: Compute $c^{\frac{1}{e}}$.
- **Relaxation 2: Small e, parts of m known**
  C '96: Ptime if $m = \tilde{m} + x$, $x < N^{\frac{1}{e}}$:

$$f(x) = (\tilde{m} + x)^e - c \bmod N$$

  For $x < N^{\frac{1}{e}+\epsilon}$: $x = kN^{\frac{1}{e}} + x'$, $k \leq N^{\epsilon}$, $x' \leq N^{\frac{1}{e}}$

RSA Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

Given: $N = pq$, $e \in \mathbb{Z}^*_{\phi(N)}$ and $c = m^e \bmod N$

Find: $m \in \mathbb{Z}_N$

- **Relaxation 1: Small e, m**
  Ptime if $m < N^{\frac{1}{e}}$: Compute $c^{\frac{1}{e}}$.
- **Relaxation 2: Small e, parts of m known**
  C '96: Ptime if $m = \tilde{m} + x$, $x < N^{\frac{1}{e}}$:

  $$f(x) = (\tilde{m} + x)^e - c \bmod N$$

  For $x < N^{\frac{1}{e}+\epsilon}$: $x = kN^{\frac{1}{e}} + x'$, $k \leq N^{\epsilon}$, $x' \leq N^{\frac{1}{e}}$
- **Relaxation 3: Small, splittable $m < 2^k$**
  BJN '00: $m = m_1 \cdot m_2$ with $m_1 \approx m_2 < 2^{\frac{k}{2}}$.
  Check: $x^e = y^{-e} \cdot c \bmod N$, $x, y = 0, \ldots, 2^{\frac{k}{2}}$

RSA Secret Key Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

Given: $N = pq$, $e \in \mathbb{Z}^*_{\phi(N)}$

Find: d such that $ed = 1 \bmod \phi(N)$

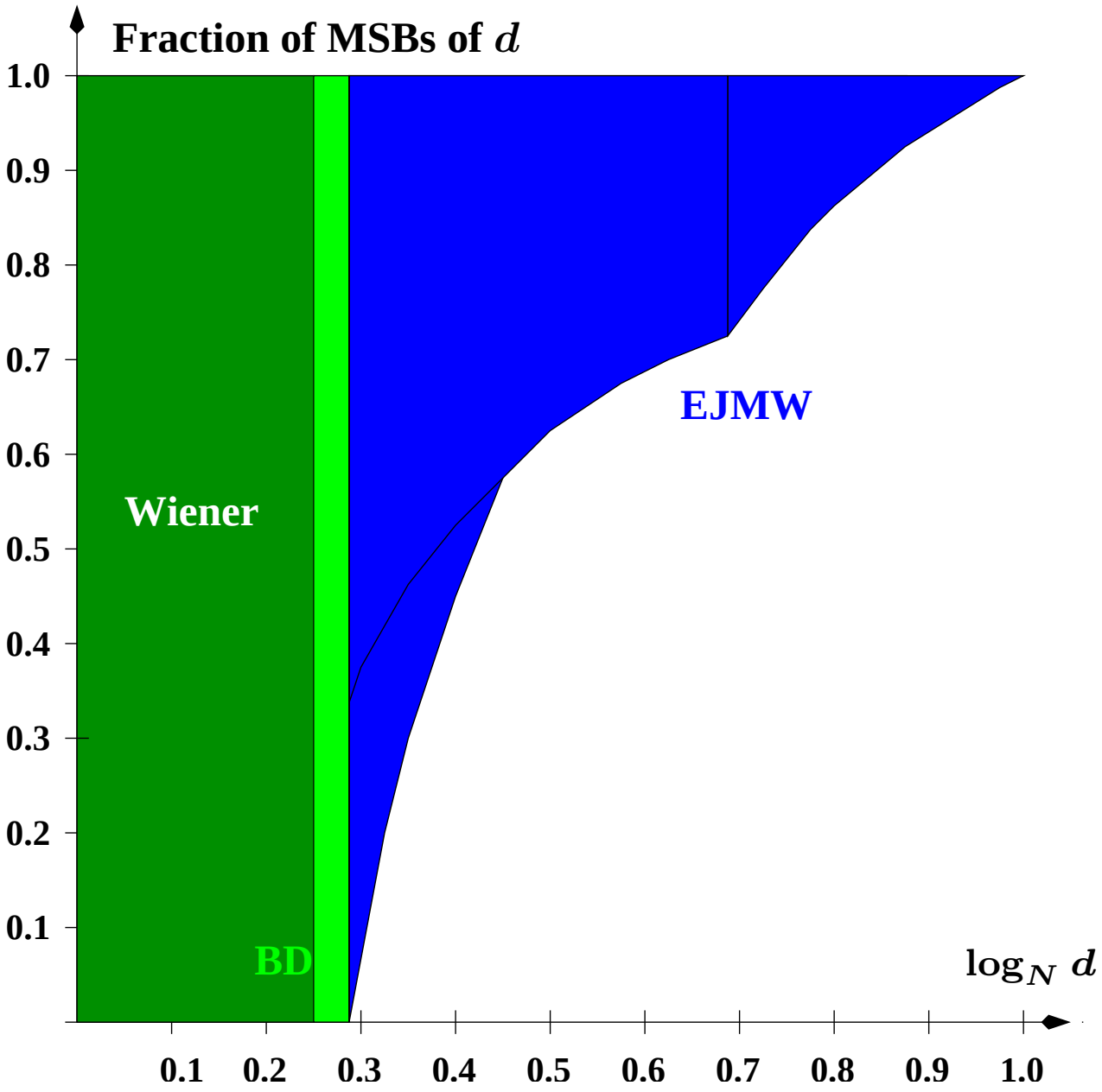
$$f(d, k, p+q) = ed + k(N + 1 - (p+q)) - 1$$

- **Relaxation 1: Small d**
  Wiener '90, $\bmod N$: $d < N^{0.25}$
  BD '99, $\bmod e$: $d < N^{0.292}$
- **Relaxation 2: Known parts of d**
  EJMW '05: Extension to full range

Open: Use properties of d: Smoothness, splittable

RSA with Small Exponent d

TECHNISCHE UNIVERSITÄT DARMSTADT



Factoring Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

Given: $N = pq$

Find: p

$$f(x) = x \bmod p$$

- **Relaxation 1: Known parts of p**
  C '96: $f(x) = \tilde{p} + x \bmod p$ with $x \leq N^{\frac{1}{4}}$.
- **Relaxation 2: "Small" p, q, i. e. $N = p^r q$**
  BDH '00: Solvable if $p = \tilde{p} + x$ with $x \leq N^{\frac{r}{(r+1)^2}}$.

Open Problems:

- Factor $N = p^r q^s$, $r \approx s$ with less bits.
- Factor $N = pqr$ with less bits.
- Factor using non-consecutive bits.

Factoring with Known Bits

TECHNISCHE UNIVERSITÄT DARMSTADT

- FOR $i = 1$ TO $N^{\frac{1}{4}}$
  - Set $\tilde{p}_i = i \cdot N^{\frac{1}{4}}$.
  - Coppersmith's method: L^3-reduce basis $B(\tilde{p}_i)$

$$B_{red}(\tilde{p}_i) = U_i \cdot B(\tilde{p}_i).$$

  - If p is found, STOP.

Complexity: $\mathcal{O}(N^{\frac{1}{4}} \cdot T(L^3\text{-reduction}))$

Open problem 1: Reduce number of guesses.

Analyze: How good is $U_i \cdot B(\tilde{p}_{i+1})$?

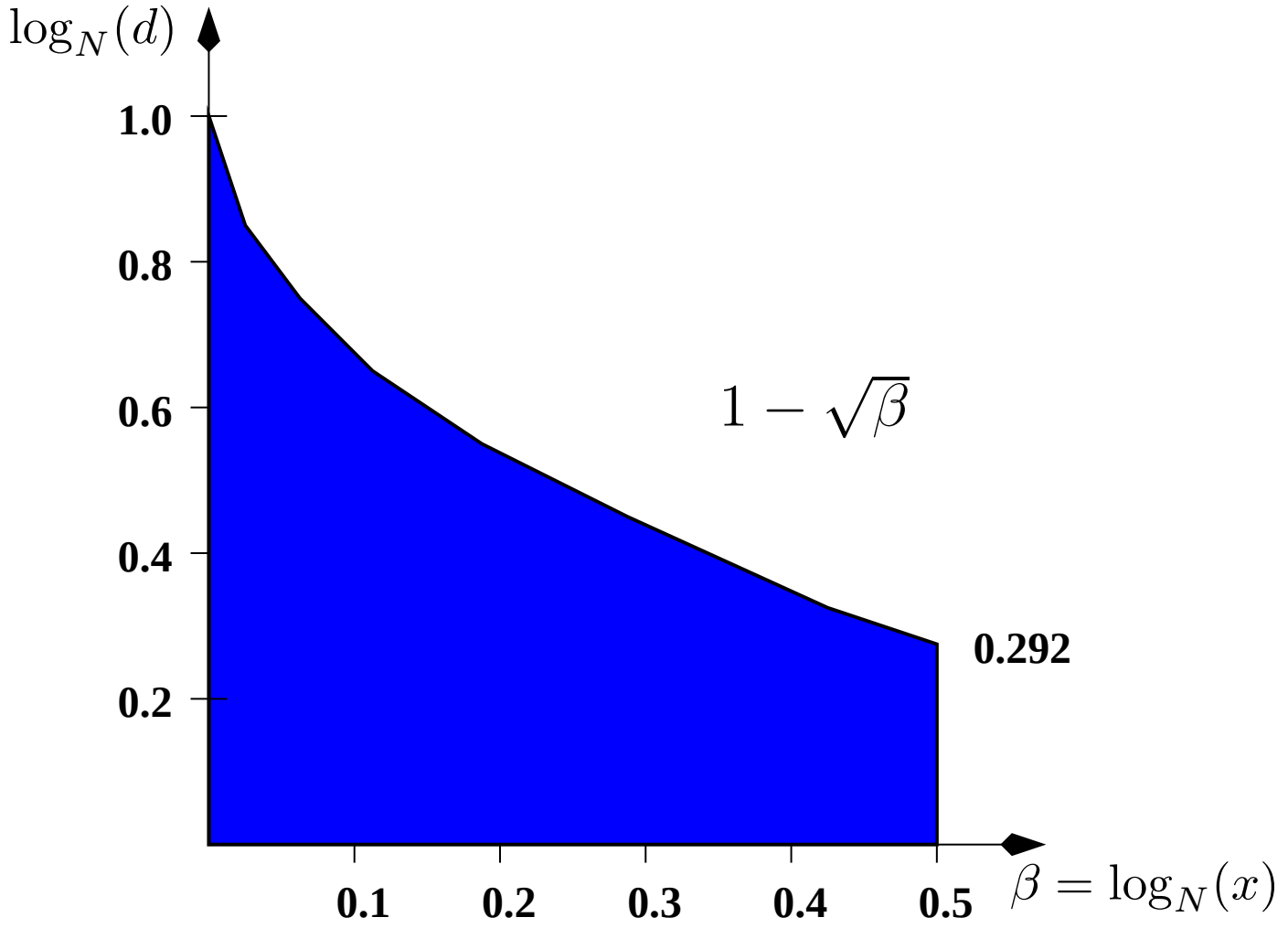
Open problem 2: Information from wrong guesses?

Combining relaxations

TECHNISCHE UNIVERSITÄT DARMSTADT

Small d and known bits of $p = \tilde{p} + x$.

$$f(k, (p+q)) = k(N + 1 - (p+q)) - 1 \bmod e$$

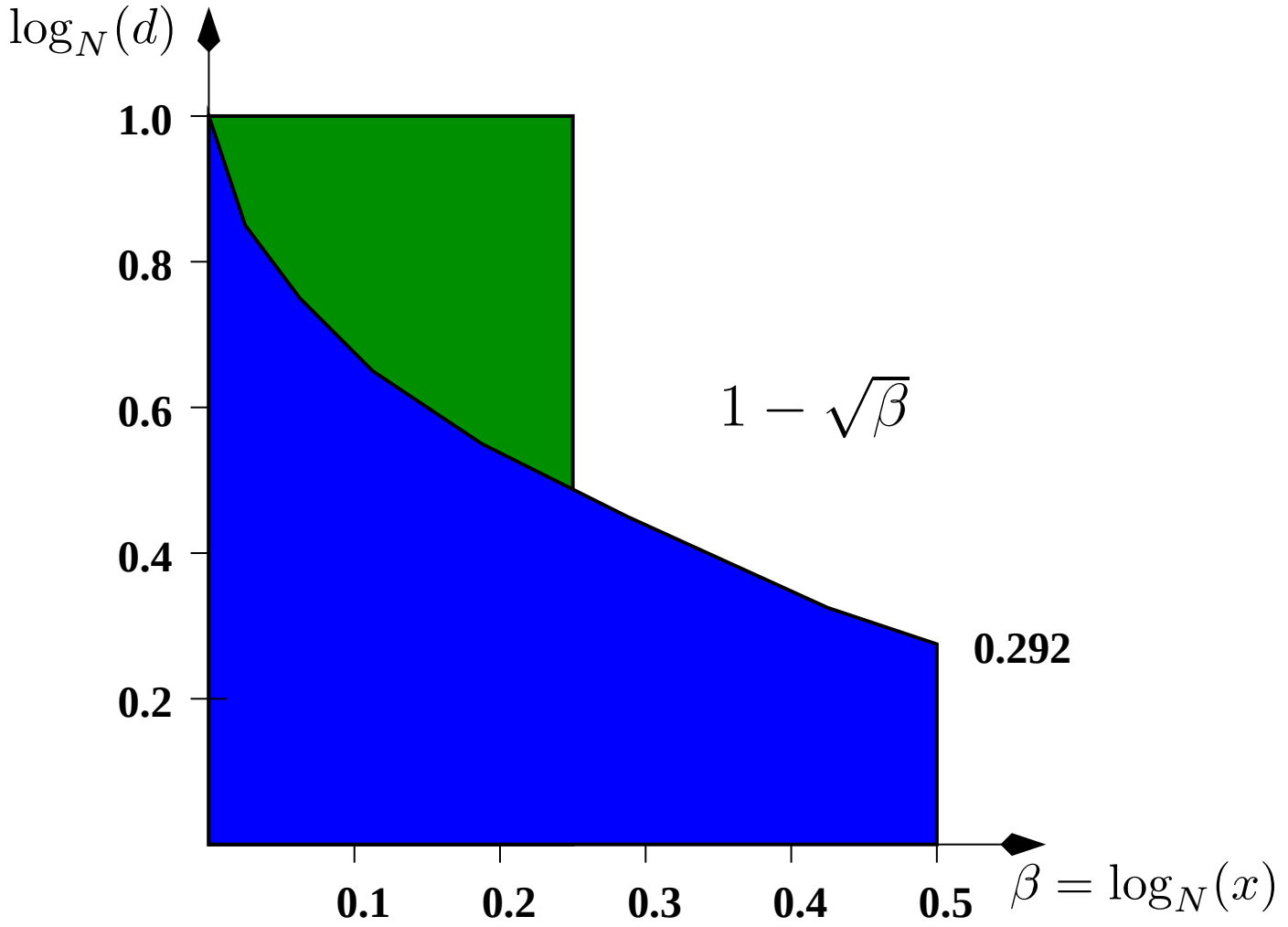


Combining relaxations

TECHNISCHE UNIVERSITÄT DARMSTADT

Small d and known bits of $p = \tilde{p} + x$.

$$f(k, (p+q)) = k(N + 1 - (p+q)) - 1 \bmod e$$



CRT-RSA problem

TECHNISCHE UNIVERSITÄT DARMSTADT

Setting:
$d = (d_p \bmod p - 1, d_q \bmod q - 1)$ with small d_p, d_q.

- **Relaxation 1: Imbalanced p, q**
  M '02: Works for $q < N^{0.382}$
  BM '06: Attack for $q < N^{0.468}$
- **Relaxation 2: Small e**
  BM '06: Cryptanalysis of two RSA-variants

For standard RSA case still $\mathcal{O}(\sqrt{\min\{d_p, d_q\}})$.

Upcoming soon (together with Ellen Jochemsz):

$$d_p, d_q \leq N^{0.073}.$$

Modular square roots

TECHNISCHE UNIVERSITÄT DARMSTADT

Given: prime p, $a \in \mathbb{Z}_p^*$ with $a^{\frac{p-1}{2}} = 1 \bmod p$.

Find: x such that $x^2 = a \bmod p$ in *deterministic* ptime without any conjecture (ERH).

$$f(x) = x^2 - a \bmod p$$

- Coppersmith's method: $|x| \leq p^{\frac{1}{2}}$
- Use group order
- Use "sparseness" of polynomial
- Use special type of solution: $x_1 = p - x_0$

Alternatively find a non-residue: Any root of

$$f(y) = y^{\frac{p-1}{2}} + 1 \bmod p.$$

Open question: Solve $f(x) = x^2 - 1 \bmod N$.

Methods: Open problems

TECHNISCHE UNIVERSITÄT DARMSTADT

- Modular Roots:
  - Coppersmith ’96
  - HG ’97: Alternative with dual lattice
- Integer Roots:
  - Coppersmith ’96
  - Coron ’04: Alternative with dual lattice

  Two open problems remain:
  - Running time
  - Shape of polynomial

Modular method

Given: $f(x)$, $N \in \mathbb{N}$ of unknown factorization

Find: Roots $|x_0| \leq X$ s.t. $f(x_0) = 0 \bmod N$.

Idea:

- Collection of polynomials $f_1(x), f_2(x), \ldots, f_n(x)$ with the roots $|x_0| \leq X$ modulo N^m.
- Construct $g(x) = \sum_{i=1}^{n} a_i f_i$, $a_i \in \mathbb{Z}$ such that

$$g(x_0) = 0 \text{ over } \mathbb{Z}.$$

Sufficient condition: $|g(x_0)| < N^m$.

- Solve $g(x)$ over the integers.

Extensions to more variables

TECHNISCHE UNIVERSITÄT DARMSTADT

Goal: Find small roots of $f(x_1, \ldots, x_n) \bmod N$.

- Compute $g_1, g_2, \ldots, g_n$ with small roots over $\mathbb{Z}$.
- Eliminate variables by resultant computations.

Open problems:

- Elimination requires algebraically independent g_i.
- Finding optimal collection:
  Complex combinatorial optimization problem.

Finding good bounds

TECHNISCHE UNIVERSITÄT DARMSTADT

Given : Polynomial $f(x_1, \ldots, x_n) (\bmod N)$

Find : Optimal bounds $|x_1| \leq X_1, \ldots, |x_n| \leq X_n$

Equiv. : Find optimal lattice basis.

Status quo:

- C '96: Solved for univariate modular case
- BM '05: Toolbox for bivariate integer case
- JM '06: Strategies for multivariate case

Work in progress (with Damien Stehlé):

- Algorithm that outputs optimal bound
- Uses Gröbner bases-like approach
- LLL reduction for selecting sub-lattice

Integer method

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\begin{aligned} f(x,y) &= (\tilde{p}+x)(\tilde{q}+y) - N \\ &= xy + \tilde{p}y + \tilde{q}x + \tilde{p}\tilde{q} - N \end{aligned}$$

Coppersmith-type lattice basis B:

$$\begin{pmatrix} \frac{1}{XY} & 0 & 0 & 1 \\ 0 & \frac{1}{Y} & 0 & \tilde{p} \\ 0 & 0 & \frac{1}{X} & \tilde{q} \\ 0 & 0 & 0 & \tilde{p}\tilde{q} - N \end{pmatrix}$$

Target vector: $(x_0 y_0, y_0, x_0, 1) \cdot B = \left(\frac{x_0 y_0}{XY}, \frac{y_0}{Y}, \frac{x_0}{X}, 0\right)$.

Integer method

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\begin{aligned} f(x,y) &= (\tilde{p}+x)(\tilde{q}+y) - N \\ &= xy + \tilde{p}y + \tilde{q}x + \tilde{p}\tilde{q} - N \end{aligned}$$

Coppersmith-type lattice basis B:

$$\begin{pmatrix} \frac{1}{XY} & 0 & 0 & 1 \\ 0 & \frac{1}{Y} & 0 & \tilde{p} \\ 0 & 0 & \frac{1}{X} & \tilde{q} \\ 0 & 0 & 0 & \tilde{p}\tilde{q} - N \end{pmatrix} \rightarrow \left(\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ \hline b_1 & b_2 & b_3 & 1 \end{array}\right)$$

Target vector: $(x_0y_0, y_0, x_0, 1) \cdot B = \left(\frac{x_0y_0}{XY}, \frac{y_0}{Y}, \frac{x_0}{X}, 0\right)$.

Coron's methods

Transform f into:

$$f'(x,y) = 1 + ax + by + cxy \mod R.$$

$$B' = \begin{pmatrix} 1 & aX & bY & cXY \\ 0 & RX & 0 & 0 \\ 0 & 0 & RY & 0 \\ 0 & 0 & 0 & RXY \end{pmatrix}$$

Vector with sufficiently small norm yields $g(x,y)$ with

$$g(x_0, y_0) = 0 \quad \text{over } \mathbb{Z}.$$

Open question: Is dimension reduction possible?

Shape Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

In case of zero constant term, e.g.

$$f(x,y) = ax^2y + bx^2 + cy^2.$$

Coron's suggestion: Transform to

$$f^*(x,y) = f(x+i,y) \quad \text{for some small } i$$

- New set of monomials: $x^2y, xy, y, x^2, x, 1, y^2$.
- Change of Newton polytope.

Problem: Analysis depends on Newton polytope.

Open problem: Fix the zero constant case.

Use bounds constructively

Assume you found for $f(x, y)$ an "optimal" bound

$$XY \leq N^{\frac{1}{2}}.$$

- Define hard problem: Find roots with

$$XY > N^{\frac{1}{2}+\delta}.$$

- Construct primitives based on this hardness.
- SPW '06: RSA-PRNG with $\Theta(\log N)$ bits output instead of $\Theta(\log \log N)$.

Open problem: Find more primitives.

Cyclic Lattices & NTRU

TECHNISCHE UNIVERSITÄT DARMSTADT

NTRU parameters:

Secret: $f, g \in \mathbb{Z}_q[x]/(x^n - 1)$, f, g small norm

Public: $h \in \mathbb{Z}_q[x]/(x^n - 1)$ with $f * h = g$

- Identify polynomial

$$f_0 + f_1 x + \ldots + f_{n-1} x^{n-1} \quad \text{with}$$
$$(f_0, f_1, \ldots, f_{n-1}).$$

- Multiplication with x rotates coefficient vector:

$$(f_0, f_1, \ldots, f_{n-1}) \mapsto (f_{n-1}, f_0, \ldots, f_{n-2})$$

Attack with Lattices

CS ’97: Lattice basis

$$B = \left(\begin{array}{cccc|cccc} 1 & 0 & \dots & 0 & h_0 & h_1 & \dots & h_{n-1} \\ 0 & 1 & \dots & 0 & h_{n-1} & h_0 & \dots & h_{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & h_1 & h_2 & \dots & h_0 \\ \hline 0 & 0 & \dots & 0 & q & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & q \end{array}\right)$$

Target vector: $t = (f_0, \dots, f_{n-1}, g_0, \dots, g_{n-1})$

Rotations of target

Lattice $L(B)$ contains also all the rotations:

$$t^i = (f_i, f_{i+1} \ldots, f_{i-1}, g_i, g_{i+1}, \ldots, g_{i-1})$$

Algorithm:

- Reduce B. Let v be a shortest vector.
- Include all rotations v^i in B and iterate.

Open Problem 1: How good is this?

Open Problem 2: Speed up lattice reduction.

Summary of Main Problems

- Complexity of Factoring with Known Bits
- Combination of known results for RSA attacks
- Automatic bound computation
- Formulation for integer case as dual lattice
- Construct primitives on “small roots assumption”
- Attack cyclic lattices