LLL algorithm and usage in cryptography

Ariana

libgen/scihub

May 16, 2020

Table of Contents

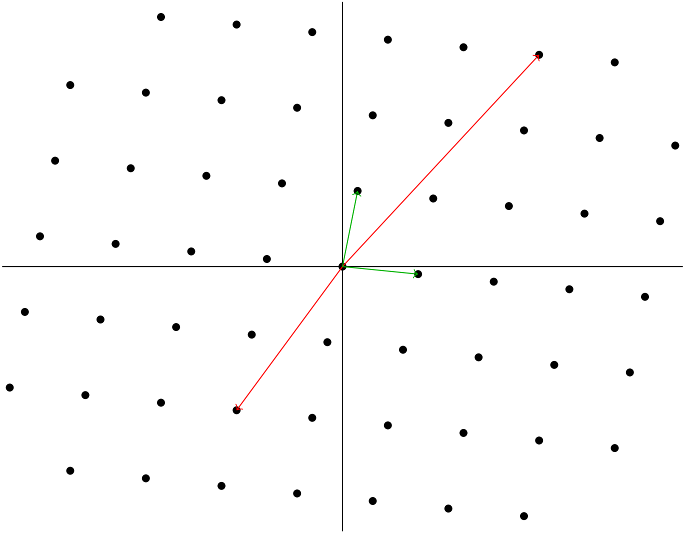
Lattices

A lattice is a free $\mathbb{Z}$-module with d generators as a subset of $\mathbb{R}^n$
Example: $\mathbb{Z}^n$ in $\mathbb{R}^n$

Lattices

A lattice is a free $\mathbb{Z}$-module with d generators as a subset of $\mathbb{R}^n$
Example: $\mathbb{Z}^n$ in $\mathbb{R}^n$
A lattice reduction algorithm is an algorithm that finds a 'short' and 'nearly orthogonal' basis

Lattice in $\mathbb{R}^2$



Euclidean algorithm

The Euclidean algorithm returns the gcd of a, b

while $b \neq 0$ **do**
  if $|a| > |b|$ **then**
    $a, b \leftarrow b, a$
  end if
  $d \leftarrow \frac{b}{a}$
  $b \leftarrow b - \lfloor d \rceil a$
end while
return a

a, b is just a lattice in $\mathbb{R}^1$ and $\gcd(a, b)$ is it's reduced lattice

Gaussian Lattice Reduction

Let $\mathbf{b}_1, \mathbf{b}_2$ be a basis

while $\lfloor \mu_{2,1} \rceil \neq 0$ **do**

  if $||\mathbf{b}_1|| > ||\mathbf{b}_2||$ **then**

    $\mathbf{b}_1, \mathbf{b}_2 \leftarrow \mathbf{b}_2, \mathbf{b}_1$

  end if

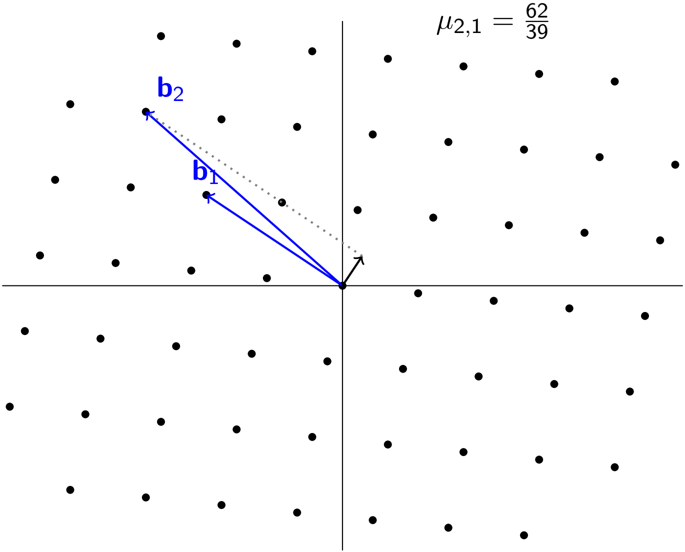
  $\mu_{2,1} \leftarrow \frac{(\mathbf{b}_2, \mathbf{b}_1)}{||b_1||^2}$

  $\mathbf{b}_2 \leftarrow \mathbf{b}_2 - \lfloor \mu_{2,1} \rceil \mathbf{b}_1$

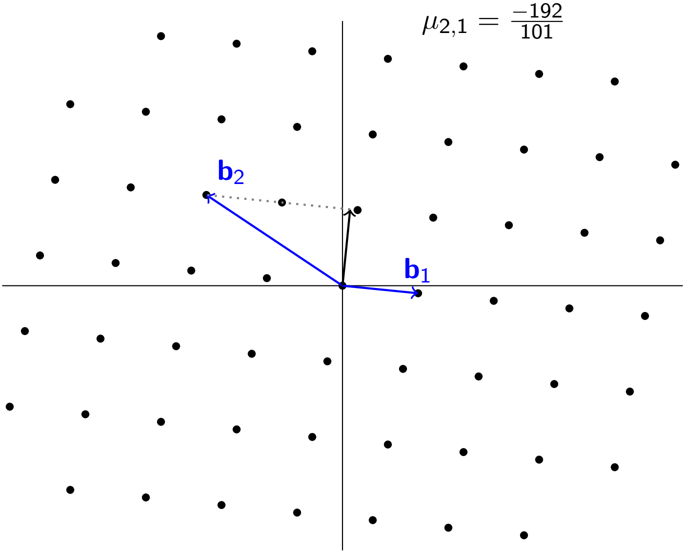
end while

return $\mathbf{b}_1, \mathbf{b}_2$

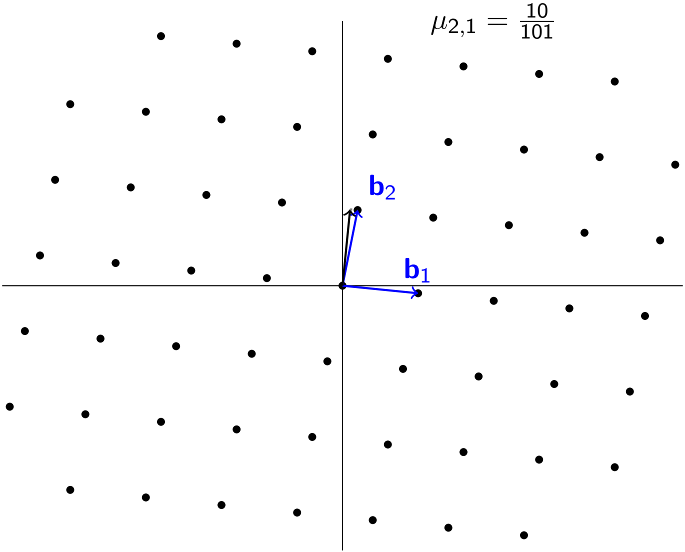
Example



Example



Example



Gram-Schmidt

For some vectors $\mathbf{b}_i \in \mathbb{R}^n$, define the orthogonal vectors $\mathbf{b}_i^*$ as

$$\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \frac{\left(\mathbf{b}_i, \mathbf{b}_j^*\right)}{\left|\left|\mathbf{b}_j^*\right|\right|^2} \mathbf{b}_j^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{j,i} \mathbf{b}_j^*$$

with $\mu_{i,j} = \frac{\left(\mathbf{b}_i, \mathbf{b}_j^*\right)}{\left|\left|\mathbf{b}_j^*\right|\right|^2}$

Then the space generated by b_i and b_i^* are the same

Typically we normalize the vectors but for lattice reduction purposes this is not done

LLL-reduced

For some basis $\mathbf{b}_i$, let $\mathbf{b}_i^*$ be the Gram-Schmidt orthogonalized basis. Then the basis is LLL-reduced for $\delta \in \left(\frac{1}{4}, 1\right)$ iff:

LLL-reduced

For some basis $\mathbf{b}_i$, let $\mathbf{b}_i^*$ be the Gram-Schmidt orthogonalized basis. Then the basis is LLL-reduced for $\delta \in \left(\frac{1}{4}, 1\right)$ iff:

$$\mu_{i,j} = \frac{\left(\mathbf{b}_i, \mathbf{b}_j^*\right)}{\left|\left|\mathbf{b}_j^*\right|\right|^2}$$

1. Size reduced: $j < i, \mu_{i,j} \leq \frac{1}{2}$

LLL-reduced

For some basis $\mathbf{b}_i$, let $\mathbf{b}_i^*$ be the Gram-Schmidt orthogonalized basis. Then the basis is LLL-reduced for $\delta \in \left(\frac{1}{4}, 1\right)$ iff:

$$\mu_{i,j} = \frac{\left(\mathbf{b}_i, \mathbf{b}_j^*\right)}{\left|\left|\mathbf{b}_j^*\right|\right|^2}$$

1. Size reduced: $j < i, \mu_{i,j} \leq \frac{1}{2}$
2. Lovász condition: $\left(\delta - \mu_{i+1,i}^2\right) ||\mathbf{b}_i^*||^2 \leq \left|\left|\mathbf{b}_{i+1}^*\right|\right|^2$

LLL algorithm

$i \leftarrow 2$
while $i < n$ **do**
  for $j = i-1, i-2, ..., 1$ **do**
    if $|\mu_{i,j}| > \frac{1}{2}$ **then**
      $\mathbf{b}_i \leftarrow \mathbf{b}_i - \lfloor \mu_{i,j} \rceil \mathbf{b}_j$
    end if
  end for
  if $\left(\delta - \mu_{i,i-1}^2\right) ||\mathbf{b}_{i-1}^*||^2 \leq ||\mathbf{b}_i^*||^2$ **then**
    $i \leftarrow i+1$
  else
    $i \leftarrow \max(i-1, 2)$
    $\mathbf{b}_{i-1}, \mathbf{b}_i \leftarrow \mathbf{b}_i, \mathbf{b}_{i-1}$
  end if
end while

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1,2,0)$	$(1,3,2)$	$(2,2,1)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1,2,0)$	$\left(-\frac{2}{5},\frac{1}{5},2\right)$	$\left(\frac{20}{21},-\frac{10}{21},\frac{5}{21}\right)$

$$\mu_{2,1} = \frac{7}{5}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1,2,0)$	$(0,1,2)$	$(2,2,1)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1,2,0)$	$\left(-\frac{2}{5},\frac{1}{5},2\right)$	$\left(\frac{20}{21},-\frac{10}{21},\frac{5}{21}\right)$

$$\mu_{2,1} = \frac{2}{5}$$

$$\left(\frac{3}{4} - \left(\frac{2}{5}\right)^2\right) ||\mathbf{b}_1^*||^2 \leq ||\mathbf{b}_2^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 2, 0)$	$(0, 1, 2)$	$(2, 2, 1)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 2, 0)$	$\left(-\frac{2}{5}, \frac{1}{5}, 2\right)$	$\left(\frac{20}{21}, -\frac{10}{21}, \frac{5}{21}\right)$

$$\mu_{3,2} = \frac{8}{21}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 2, 0)$	$(0, 1, 2)$	$(2, 2, 1)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 2, 0)$	$\left(-\frac{2}{5}, \frac{1}{5}, 2\right)$	$\left(\frac{20}{21}, -\frac{10}{21}, \frac{5}{21}\right)$

$$\mu_{3,1} = \frac{6}{5}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 2, 0)$	$(0, 1, 2)$	$(1, 0, 1)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 2, 0)$	$\left(-\frac{2}{5}, \frac{1}{5}, 2\right)$	$\left(\frac{20}{21}, -\frac{10}{21}, \frac{5}{21}\right)$

$$\mu_{3,2} = \frac{8}{21}$$

$$\left(\frac{3}{4} - \left(\frac{8}{21}\right)^2\right) ||\mathbf{b}_2^*||^2 > ||\mathbf{b}_3^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 2, 0)$	$(1, 0, 1)$	$(0, 1, 2)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 2, 0)$	$\left(\frac{4}{5}, -\frac{2}{5}, 1\right)$	$\left(-\frac{10}{9}, -\frac{5}{9}, \frac{10}{9}\right)$

$$\mu_{2,1} = \frac{1}{5}$$

Example

$$\begin{array}{ccc} \mathbf{b}_1 & \mathbf{b}_2 & \mathbf{b}_3 \\ (1,2,0) & (1,0,1) & (0,1,2) \end{array}$$

$$\begin{array}{ccc} \mathbf{b}_1^* & \mathbf{b}_2^* & \mathbf{b}_3^* \\ (1,2,0) & \left(\frac{4}{5},-\frac{2}{5},1\right) & \left(-\frac{10}{9},-\frac{5}{9},\frac{10}{9}\right) \end{array}$$

$$\mu_{2,1} = \frac{1}{5}$$

$$\left(\frac{3}{4} - \left(\frac{1}{5}\right)^2\right) ||\mathbf{b}_1^*||^2 > ||\mathbf{b}_2^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(1, 2, 0)$	$(0, 1, 2)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$\left(\frac{1}{2}, 2, -\frac{1}{2}\right)$	$\left(-\frac{10}{9}, -\frac{5}{9}, \frac{10}{9}\right)$

$$\mu_{2,1} = \frac{1}{2}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(1, 2, 0)$	$(0, 1, 2)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$\left(\frac{1}{2}, 2, -\frac{1}{2}\right)$	$\left(-\frac{10}{9}, -\frac{5}{9}, \frac{10}{9}\right)$

$$\mu_{2,1} = \frac{1}{2}$$

$$\left(\frac{3}{4} - \left(\frac{1}{2}\right)^2\right) ||\mathbf{b}_1^*||^2 \leq ||\mathbf{b}_2^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1,0,1)$	$(1,2,0)$	$(0,1,2)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1,0,1)$	$\left(\frac{1}{2},2,-\frac{1}{2}\right)$	$\left(-\frac{10}{9},-\frac{5}{9},\frac{10}{9}\right)$

$$\mu_{3,2} = \frac{2}{9}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(1, 2, 0)$	$(0, 1, 2)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$\left(\frac{1}{2}, 2, -\frac{1}{2}\right)$	$\left(-\frac{10}{9}, -\frac{5}{9}, \frac{10}{9}\right)$

$$\mu_{3,1} = 1$$

Example

$$\begin{array}{ccc} \mathbf{b}_1 & \mathbf{b}_2 & \mathbf{b}_3 \\ (1,0,1) & (1,2,0) & (-1,1,1) \end{array}$$

$$\begin{array}{ccc} \mathbf{b}_1^* & \mathbf{b}_2^* & \mathbf{b}_3^* \\ (1,0,1) & \left(\frac{1}{2},2,-\frac{1}{2}\right) & \left(-\frac{10}{9},-\frac{5}{9},\frac{10}{9}\right) \end{array}$$

$$\mu_{3,2} = \frac{2}{9}$$

$$\left(\frac{3}{4} - \left(\frac{2}{9}\right)^2\right) ||\mathbf{b}_2^*||^2 > ||\mathbf{b}_3^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(-1, 1, 1)$	$(1, 2, 0)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$(-1, 1, 1)$	$\left(\frac{5}{6}, -\frac{5}{3}, -\frac{5}{6}\right)$

$$\mu_{2,1} = 0$$

Example

$$\begin{array}{ccc} \mathbf{b}_1 & \mathbf{b}_2 & \mathbf{b}_3 \\ (1,0,1) & (-1,1,1) & (1,2,0) \end{array}$$

$$\begin{array}{ccc} \mathbf{b}_1^* & \mathbf{b}_2^* & \mathbf{b}_3^* \\ (1,0,1) & (-1,1,1) & \left(\frac{5}{6},-\frac{5}{3},-\frac{5}{6}\right) \end{array}$$

$$\mu_{2,1} = 0$$

$$\left(\frac{3}{4} - (0)^2\right) ||\mathbf{b}_1^*||^2 \leq ||\mathbf{b}_2^*||^2$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(-1, 1, 1)$	$(1, 2, 0)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$(-1, 1, 1)$	$\left(\frac{5}{6}, -\frac{5}{3}, -\frac{5}{6}\right)$

$$\mu_{3,2} = \frac{1}{3}$$

Example

$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1, 0, 1)$	$(-1, 1, 1)$	$(1, 2, 0)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1, 0, 1)$	$(-1, 1, 1)$	$\left(\frac{5}{6}, -\frac{5}{3}, -\frac{5}{6}\right)$

$$\mu_{3,1} = \frac{1}{2}$$

Example

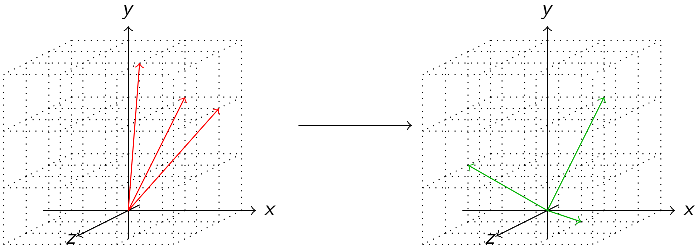
$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$
$(1,0,1)$	$(-1,1,1)$	$(1,2,0)$
$\mathbf{b}_1^*$	$\mathbf{b}_2^*$	$\mathbf{b}_3^*$
$(1,0,1)$	$(-1,1,1)$	$\left(\frac{5}{6},-\frac{5}{3},-\frac{5}{6}\right)$

$$\mu_{3,1} = \frac{1}{2}$$

$$\left(\frac{3}{4} - \left(\frac{1}{2}\right)^2\right) ||\mathbf{b}_2^*||^2 \leq ||\mathbf{b}_3^*||^2$$

Example

$$\begin{pmatrix} 1 & 2 & 0 \\ 1 & 3 & 2 \\ 2 & 2 & 1 \end{pmatrix} \to \begin{pmatrix} 1 & 0 & 1 \\ -1 & 1 & 1 \\ 1 & 2 & 0 \end{pmatrix}$$



Bounds

A lattice is a free $\mathbb{Z}$-module with d generators as a subset of $\mathbb{R}^n$
Some matrix B generate a lattice with its rows as the basis b_i

$$\det(B) = \sqrt{\det\left(BB^T\right)} = \prod_i ||b_i^*||$$

Bounds

A lattice is a free $\mathbb{Z}$-module with d generators as a subset of $\mathbb{R}^n$
Some matrix B generate a lattice with its rows as the basis b_i

$$\det(B) = \sqrt{\det\left(BB^T\right)} = \prod_i ||b_i^*||$$

Suppose B is LLL-reduced and let λ_1 be length of the shortest vector in the lattice

$$||b_1|| \leq \min\left(\left(\frac{4}{4\delta - 1}\right)^{\frac{d-1}{2}} \lambda_1, \left(\frac{4}{4\delta - 1}\right)^{\frac{d-1}{4}} \det(L)^{\frac{1}{d}}\right)$$

Bounds

A lattice is a free $\mathbb{Z}$-module with d generators as a subset of $\mathbb{R}^n$
Some matrix B generate a lattice with its rows as the basis b_i

$$\det(B) = \sqrt{\det\left(BB^T\right)} = \prod_i ||b_i^*||$$

Suppose B is LLL-reduced and let λ_1 be length of the shortest vector in the lattice

$$||b_1|| \leq \min\left(\left(\frac{4}{4\delta - 1}\right)^{\frac{d-1}{2}} \lambda_1, \left(\frac{4}{4\delta - 1}\right)^{\frac{d-1}{4}} \det(L)^{\frac{1}{d}}\right)$$

For random lattices LLL usually finds $||b_1|| \lesssim 1.02^d \det(L)^{\frac{1}{d}}$

Rational approximation

To find a rational approximation of x, let B be a big number.

$$\begin{pmatrix} 1 & 0 & xB \\ 0 & 1 & -B \end{pmatrix}$$

Smallest vector from LLL is of the form (a, b, k) with
$0 \approx \frac{k}{B} = ax - b$

Approximate integer linear relations

Let x_i be some arbitrary numbers and B be a big number

$$\begin{pmatrix} 1 & 0 & \dots & 0 & x_1 B \\ 0 & 1 & \dots & 0 & x_2 B \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & x_n B \end{pmatrix}$$

Smallest vector from LLL is of the form $(c_1, c_2, \dots, c_n, x)$ with $\sum c_i x_i \approx 0$

Algebraic number approximation

To find a algebraic approximation of x, let B be a big number and n be the degree of a polynomial

$$\begin{pmatrix} 1 & 0 & \dots & 0 & B \\ 0 & 1 & \dots & 0 & xB \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & x^n B \end{pmatrix}$$

Then the smallest vector of the LLL reduced matrix is of the form $(f_0, f_1, \dots, f_n, k)$ with k small $\sum f_i x^i \approx 0$

Howgrave Graham

Let $f(x)$ be some univariate polynomial of degree d. For some modulus N and bound B:
$f(x_0) = 0 \pmod N$, $x_0 < B$ and $|f(x)| < N$ for all $0 < x < B$ implies $f(x_0) = 0$ over $\mathbb{R}$

Howgrave Graham

Let $f(x)$ be some univariate polynomial of degree d. For some modulus N and bound B:
$f(x_0) = 0 \pmod N$, $x_0 < B$ and $|f(x)| < N$ for all $0 < x < B$ implies $f(x_0) = 0$ over $\mathbb{R}$
$f(x_0) = 0 \pmod N$ and $||f(Bx)||_2 < \frac{N}{\sqrt{d}}$ implies $f(x_0) = 0$ over $\mathbb{R}$

Coppersmith algorithm(sketch)

If $x_0 < B$ is a root for some polynomials f, g_i in $\frac{\mathbb{Z}}{N\mathbb{Z}}$, then the lattice generated by f, g_i all have x_0 as a root in $\frac{\mathbb{Z}}{N\mathbb{Z}}$

Coppersmith algorithm(sketch)

If $x_0 < B$ is a root for some polynomials f, g_i in $\frac{\mathbb{Z}}{N\mathbb{Z}}$, then the lattice generated by f, g_i all have x_0 as a root in $\frac{\mathbb{Z}}{N\mathbb{Z}}$

1. Construct polynomials g_i
2. Use $f(Bx)$ and $g_i(Bx)$ in the lattice
3. h is hopefully a small vector in the lattice with $||h(x)||_2 < \frac{N}{\sqrt{d}} \implies h\left(\frac{x_0}{B}\right) = 0$ in $\mathbb{R}$

Coppersmith algorithm

$g_i(x) = Nx^i$ is has root x_0 in $\frac{\mathbb{Z}}{N\mathbb{Z}}$

$$G = \begin{pmatrix} N & 0 & 0 & \dots & 0 & 0 \\ 0 & NB & 0 & \dots & 0 & 0 \\ 0 & 0 & NB^2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & NB^{d-1} & 0 \\ f_0 & f_1 B & f_2 B^2 & \dots & f_{d-1}B^{d-1} & B^d \end{pmatrix}$$

$$\det(G) = N^d B^{\frac{d(d+1)}{2}} \quad \dim(G) = d + 1$$

Let $\mathbf{v}$ be a short vector from LLL, then $h(x) = \sum_{i=0}^{n} v_i x^i$ possibly has a root $\frac{x_0}{B}$ over $\mathbb{R}$

Theoretical discussion

Current lattice only ensures shortest vector of $O\left(N^{\frac{d}{d+1}}B^{\frac{d}{2}}\right)$, which must be less than $O(N)$ to work, so $B < O\left(N^{\frac{2}{d(d+1)}}\right)$

$B < N^{\frac{1}{d}}$ is a open conjectured theoretical limit for finding 'small roots' efficiently
Take $f(x) = x^2 + px \pmod{p}^2$, if $B = p^{\frac{1}{d}+\epsilon}$, number of small roots is unbounded and our polynomial over integers can't have so many roots

Add more vectors in $(f(x), N)$ to decrease $\det(G)^{\frac{1}{d}}$

Notation

Let g_i be some polynomials $\sum_j g_{i,j} x^j$, then define the lattice G generated from these polynomials as

$$G = \begin{pmatrix} g_{0,0} & g_{0,1} & g_{0,2} & \cdots \\ g_{1,0} & g_{1,1} & g_{1,2} & \cdots \\ g_{2,0} & g_{2,1} & g_{2,2} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

First improvement

Define $g_{0,j}(x) = Nx^j$ and $g_{1,j}(x) = f(x)x^j$, $0 \leq j < d$ and construct a lattice G using coefficients of $g_{i,j}(Bx)$

$$\det(G) = N^d B^{\frac{(2d-1)2d}{2}} \quad \dim(G) = 2d$$

The shortest vector has length $O\left(N^{\frac{1}{2}} B^{\frac{2d-1}{2}}\right)$, bounded by $O(N)$ to find small roots

$$B < O\left(N^{\frac{1}{2d-1}}\right)$$

Some motivation

$f(x)^a \pmod{N^a}$ has the same roots as $f(x) \pmod{N}$

Some motivation

$f(x)^a \pmod{N^a}$ has the same roots as $f(x) \pmod{N}$
$N^a g(x) \pmod{N^{a+b}}$ has the same roots as $g(x) \pmod{N^b}$

Some motivation

$f(x)^a \pmod{N^a}$ has the same roots as $f(x) \pmod{N}$
$N^a g(x) \pmod{N^{a+b}}$ has the same roots as $g(x) \pmod{N^b}$
Adding more vectors(strategically) decreases $\frac{N^m}{\det(L)^{\frac{1}{d}}}$, allowing for larger bounds of size of roots

Final improvement

Define $g_{i,j}(x) = N^{h-j} f(x)^j x^i$ for some h, $0 \le i < d$, $0 \le j < h$ and construct a lattice G using coefficients of $g_{i,j}(Bx)$

$$\det(G) = N^{\frac{dh(h+1)}{2}} B^{\frac{(dh-1)dh}{2}} \quad \dim(G) = dh$$

The shortest vector has length $O\left(N^{\frac{h+1}{2}} B^{\frac{dh-1}{2}}\right)$, bounded by $O\left(N^h\right)$ to find small roots

$$B < O\left(N^{\frac{h-1}{dh-1}}\right)$$

$\lim_{h\to\infty} \frac{h-1}{dh-1} = \frac{1}{d}$, can get arbitrary close to $N^{\frac{1}{d}}$

Example

For some bound B, polynomial $x^3 + f_2 x^2 + f_1 x + f_2$ and modulus N
$h = 3$, $g_{i,j}(x) = N^{h-j} f(x)^j x^i$, $0 \le i < d$, $0 \le j < h$

$$\begin{pmatrix}
N^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & BN^3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & B^2 N^3 & 0 & 0 & 0 & 0 & 0 & 0 \\
N^2 f_0 & BN^2 f_1 & B^2 N^2 f_2 & B^3 N^2 & 0 & 0 & 0 & 0 & 0 \\
0 & BN^2 f_0 & B^2 N^2 f_1 & B^3 N^2 f_2 & B^4 N^2 & 0 & 0 & 0 & 0 \\
0 & 0 & B^2 N^2 f_0 & B^3 N^2 f_1 & B^4 N^2 f_2 & B^5 N^2 & 0 & 0 & 0 \\
N f_0^2 & 2\,BN f_0 f_1 & (N f_1^2 + 2\,N f_0 f_2) B^2 & 2\,(N f_1 f_2 + N f_0) B^3 & (N f_2^2 + 2\,N f_1) B^4 & 2\,B^5 N f_2 & B^6 N & 0 & 0 \\
0 & BN f_0^2 & 2\,B^2 N f_0 f_1 & (N f_1^2 + 2\,N f_0 f_2) B^3 & 2\,(N f_1 f_2 + N f_0) B^4 & (N f_2^2 + 2\,N f_1) B^5 & 2\,B^6 N f_2 & B^7 N & 0 \\
0 & 0 & B^2 N f_0^2 & 2\,B^3 N f_0 f_1 & (N f_1^2 + 2\,N f_0 f_2) B^4 & 2\,(N f_1 f_2 + N f_0) B^5 & (N f_2^2 + 2\,N f_1) B^6 & 2\,B^7 N f_2 & B^8 N
\end{pmatrix}$$

Unknown modulus

Unknown modulus $p < N^\beta$ with $p|N$

Unknown modulus

Unknown modulus $p < N^{\beta}$ with $p|N$
Define $g_{i,j}(x) = N^{h-j} f(x)^j x^i$, $0 \le i < d$, $0 \le j < h$ and $g_{i,h} = f(x)^h x^i$ with $0 \le i < t$ and construct a lattice G using coefficients of $g_{i,j}(Bx)$ and let $n = dh + t$ for convenience.

$$\det(G) = N^{\frac{dh(h+11)}{2}} B^{\frac{(n-1)n}{2}} \quad \dim(G) = n$$

The shortest vector has length $O\left(N^{\frac{dh(h+1)}{2n}} B^{\frac{n-1}{2}}\right)$, bounded by $O\left(N^{\beta h}\right)$ to find small roots

$$B < O\left(N^{\frac{n-1}{n}\left(\frac{2\beta h}{n} - \frac{dh(h+1)}{n^2}\right)}\right) \overset{n=\frac{d}{\beta}h}{=} O\left(N^{\frac{n-1}{n}\left(2-\frac{h+1}{h}\right)\frac{\beta^2}{d}}\right)$$

$$\lim_{h,n\to\infty} \frac{n-1}{n}\left(1 - \frac{1}{h}\right)\frac{\beta^2}{d} = \frac{\beta^2}{d}$$

Multivariate

Using the polynomials $g_{i,j,k,\ldots} = N^{h-i} f(x, y, \ldots)^i x^j y^k \ldots$ and $f(x)^h x^i y^j \ldots$ to construct a lattice and get polynomials with identical small roots over integers

Multivariate

Using the polynomials $g_{i,j,k,\dots} = N^{h-i} f(x, y, \dots)^i x^j y^k \dots$ and $f(x)^h x^i y^j \dots$ to construct a lattice and get polynomials with identical small roots over integers

Multivariate polynomials have infinitely many roots($x - y$) and finding integer solutions may be hard($x^2 - yN - z$ for fixed N)

Multivariate

Using the polynomials $g_{i,j,k,\ldots} = N^{h-i} f(x, y, \ldots)^i x^j y^k \ldots$ and $f(x)^h x^i y^j \ldots$ to construct a lattice and get polynomials with identical small roots over integers

Multivariate polynomials have infinitely many roots($x - y$) and finding integer solutions may be hard($x^2 - yN - z$ for fixed N)

Find simultaneous integer roots of polynomials in lattice and hope that it results in finding roots to univariate polynomials

Multivariate

Using the polynomials $g_{i,j,k,\ldots} = N^{h-i} f(x, y, \ldots)^i x^j y^k \ldots$ and $f(x)^h x^i y^j \ldots$ to construct a lattice and get polynomials with identical small roots over integers

Multivariate polynomials have infinitely many roots($x - y$) and finding integer solutions may be hard($x^2 - yN - z$ for fixed N)

Find simultaneous integer roots of polynomials in lattice and hope that it results in finding roots to univariate polynomials

Determinant is hard to compute, bound is of the form $XY \cdots < O(N^x)$ where $x < X, y < Y, \ldots$ so they can't be too big

Summary

LLL finds a short vector in a lattice

Summary

LLL finds a short vector in a lattice

Coppersmith algorithm can find small roots of univariate and bivariate polynomials mod a potentially unknown factor of N

Usage

- Finding small/short solutions
- Recovering information with noise
- Miscellaneous

Mertens conjecture and roots of $\zeta(t)$

$$|M(n)| = \left|\sum_{k=1}^{n} \mu(k)\right| < \sqrt{n}?$$

Mertens conjecture and roots of $\zeta(t)$

$$|M(n)| = \left|\sum_{k=1}^{n} \mu(k)\right| < \sqrt{n}?$$

Let ρ be the real roots of $\zeta\left(\frac{1}{2} + it\right)$, then the conjecture implies existence of infinitely many small $c_\rho \in \mathbb{Z}$ such that

$$\sum_{\rho} c_\rho \rho = 0$$

Mertens conjecture and roots of $\zeta(t)$

$$|M(n)| = \left|\sum_{k=1}^{n} \mu(k)\right| < \sqrt{n}?$$

Let ρ be the real roots of $\zeta\left(\frac{1}{2} + it\right)$, then the conjecture implies existence of infinitely many small $c_\rho \in \mathbb{Z}$ such that

$$\sum_{\rho} c_\rho \rho = 0$$

Bound c_ρ assuming Mertens and with LLL on roots

$$\rho < 2516 \implies \limsup_{x\to\infty} \frac{M(x)}{\sqrt{x}} > 1.06 \quad \liminf_{x\to\infty} \frac{M(x)}{\sqrt{x}} < -1.009$$

RSA

$N = pq$ for primes p, q and e, d such that $ed = 1 \pmod{\lambda(N)}$.
Note that usually $ed = 1 \pmod{\phi(N)}$
Encryption: $c = m^e \pmod{N}$
Decryption: $m = c^d \pmod{N}$

Franklin-Reiter Related Message Attack

$m_2 = f(m_1)$, f a known polynomial and c_1, c_2 are ciphertexts of m_1, m_2
$x^e - c_1 \pmod{N}$ and $f(x)^e - c_2 \pmod{N}$ has m_1 as a root

Franklin-Reiter Related Message Attack

$m_2 = f(m_1)$, f a known polynomial and c_1, c_2 are ciphertexts of m_1, m_2
$x^e - c_1 \pmod N$ and $f(x)^e - c_2 \pmod N$ has m_1 as a root

$$\gcd_{\frac{\mathbb{Z}}{N\mathbb{Z}}[x]}(x^e - c_1, f(x)^e - c_2) = x - m_1$$

Coppersmith's Short Pad Attack

$m_2 = m_1 + r_1$ for some pad r_1, and c_1, c_2 are ciphertexts of m_1, m_2

Coppersmith's Short Pad Attack

$m_2 = m_1 + r_1$ for some pad r_1, and c_1, c_2 are ciphertexts of m_1, m_2

$$\mathrm{res}_x(f(x), g(x)) = 0 \iff f \text{ and } g \text{ shares a root}$$

Coppersmith's Short Pad Attack

$m_2 = m_1 + r_1$ for some pad r_1, and c_1, c_2 are ciphertexts of m_1, m_2

$$\mathrm{res}_x(f(x), g(x)) = 0 \iff f \text{ and } g \text{ shares a root}$$

$$f(y) = \mathrm{res}_x\left(x^e - c_1, (x+y)^e - c_2\right)$$

Find a small root of $f(y) \pmod N$ with coppersmith algorithm

Known approximation of factor

If $p_0 \approx p$, find 'small roots' of $p + x \pmod{N}$ with coppersmith algorithm

Known approximation of factor

If $p_0 \approx p$, find 'small roots' of $p + x \pmod N$ with coppersmith algorithm

$$N = pq \quad p \approx r_p t, q \approx r_q t$$

$$t \approx \sqrt{\frac{N}{r_p r_q}} \implies N = (r_p t + x)(r_q t + y)$$

Approximately similar prime factors

Assume we have modulus $N_i = p_i q_i$ with p_i close to each other, construct a lattice with columns having 2 non-zero elements, $N_i, -N_j$ and the ith row lacking $\pm N_i$
Example:

$$\begin{pmatrix} N_2 & N_3 & 0 \\ -N_1 & 0 & N_3 \\ -N_1 & -N_2 & 0 \end{pmatrix}$$

Since $q_i N_j - q_j N_i = q_i q_j (p_i - p_j)$ is small, LLL is likely to find such a vector and we can take GCD

Wiener attack

If d is small, we can compute d by simple algebraic means:

$$ed - 1 = k\phi(N) \implies \frac{e}{\phi(N)} - \frac{k}{d} = \frac{1}{d\phi(N)}$$

Wiener attack

If d is small, we can compute d by simple algebraic means:

$$ed - 1 = k\phi(N) \implies \frac{e}{\phi(N)} - \frac{k}{d} = \frac{1}{d\phi(N)}$$

$$\frac{e}{N} \approx \frac{k}{d}$$

Note that for $d < N^{\frac{1}{4}}$, $\frac{k}{d}$ is in the convergents of $\frac{e}{N}$'s continued fractions

Boneh-Durfee attack

$$ed = 1 + x(p-1)(q-1) = 1 + x(N-y) \equiv 0 \pmod{e}$$

$$d < O\left(N^{\frac{7-2\sqrt{7}}{6} \approx 0.284}\right)$$

Boneh-Durfee attack

$$ed = 1 + x(p-1)(q-1) = 1 + x(N-y) \equiv 0 \pmod{e}$$

$$d < O\left(N^{\frac{7-2\sqrt{7}}{6} \approx 0.284}\right)$$

Removing certain ‘bad vectors’:

$$d < O\left(N^{1-\frac{1}{\sqrt{2}} \approx 0.292}\right)$$

Boneh-Durfee attack

$$ed = 1 + x(p-1)(q-1) = 1 + x(N-y) \equiv 0 \pmod{e}$$

$$d < O\left(N^{\frac{7-2\sqrt{7}}{6} \approx 0.284}\right)$$

Removing certain ‘bad vectors’:

$$d < O\left(N^{1-\frac{1}{\sqrt{2}} \approx 0.292}\right)$$

$$d < O\left(N^{\frac{1}{2}}\right)?$$

Weak NTRU keys

$f, g \in \frac{\mathbb{Z}[x]}{x^N - 1}$, coefficients of f, g are $-1, 0, 1$. $f_p f = 1 \pmod{p}$ and $h = p f_p g \pmod{q}$

Weak NTRU keys

$f, g \in \frac{\mathbb{Z}[x]}{x^N - 1}$, coefficients of f, g are $-1, 0, 1$. $f_p f = 1 \pmod{p}$ and $h = p f_p g \pmod{q}$

$$L = \begin{pmatrix} \lambda I_N & 0 \\ H & q I_n \end{pmatrix}$$

where H is circulant matrix with first column being coefficients of $f_p g \pmod{q}$

$L \begin{pmatrix} f' \\ kq \end{pmatrix} = \begin{pmatrix} \lambda f' \\ g' \end{pmatrix}$ is hopefully short for some k. $pg' = f'h \pmod{q}$ breaks NTRU

Coppersmith in the wild

Primes of the form $p = a + 2^t x + y$ with a known and t bruteforcable, x, y unknown errors appeared in Taiwan's national Citizen Digital Certificate database

Coppersmith in the wild

Primes of the form $p = a + 2^t x + y$ with a known and t bruteforcable, x, y unknown errors appeared in Taiwan's national Citizen Digital Certificate database

Coppersmith method for bivariate polynomial and unknown modulus worked, but the theoretical bounds are not satisfied

ROCA attack

Primes of the form $p = kM + (e^a \pmod{M})$ with M being some primorial and $e = 65537$ was used, keys using these can be factored with coppersmith, hence the name the Return Of Coppersmith Attack

ROCA attack

Primes of the form $p = kM + (e^a \pmod{M})$ with M being some primorial and $e = 65537$ was used, keys using these can be factored with coppersmith, hence the name the Return Of Coppersmith Attack

$$N = (kM + e^a \bmod M)(lM + e^b \bmod M) \equiv e^{a+b} \pmod{M}$$

By bruteforcing a in a certain way, we can construct the polynomial $xM + (65537^a \pmod{M})$ and find small roots

References

Steven Galraith - Mathematics of Public Key Cryptography

J.W.S. Cassels - An Introduction to the Geometry of Numbers

Xinyue, D. An Introduction to Lenstra-Lenstra-Lovasz Lattice Basis Reduction Algorithm

Howgrave-Graham, N. (1997). Finding small roots of univariate modular equations revisited. Lecture Notes in Computer Science, 131–142. doi:10.1007/bfb0024458

Coppersmith D. (1996) Finding a Small Root of a Univariate Modular Equation. In: Maurer U. (eds) Advances in Cryptology — EUROCRYPT '96. EUROCRYPT 1996. Lecture Notes in Computer Science, vol 1070. Springer, Berlin, Heidelberg

Coppersmith, D. (1996). Finding a Small Root of a Bivariate Integer Equation; Factoring with High Bits Known. Lecture Notes in Computer Science, 178–189. doi:10.1007/3-540-68339-9_16

Nguyen P.Q., Stehlé D. (2006) LLL on the Average. In: Hess F., Pauli S., Pohst M. (eds) Algorithmic Number Theory. ANTS 2006. Lecture Notes in Computer Science, vol 4076. Springer, Berlin, Heidelberg

Disproof of the Mertens conjecture. (1985). Journal Für Die Reine Und Angewandte Mathematik (Crelles Journal), 1985(357). doi:10.1515/crll.1985.357.138

Jean-Charles Faugère, Raphaël Marinier, Guénaël Renault. Implicit Factoring with Shared Most Significant and Middle Bits. In 13th International Conference on Practice and Theory in PublicKey Cryptography – PKC 2010, May 2010, Paris, France. pp.70-87, 10.1007/978-3-642-13013-7_5. hal-01288914

Takayasu, A., Kunihiro, N. (2019). Partial key exposure attacks on RSA: Achieving the Boneh–Durfee bound. Theoretical Computer Science, 761, 51–77. doi: 10.1016/j.tcs.2018.08.021

Coppersmith, D., Shamir, A. (1997). Lattice Attacks on NTRU. Advances in Cryptology — EUROCRYPT '97 Lecture Notes in Computer Science, 52–61. doi: 10.1007/3-540-69053-0_5

Bernstein, D. J., Chang, Y.-A., Cheng, C.-M., Chou, L.-P., Heninger, N., Lange, T., Someren, N. V. (2013). Factoring RSA Keys from Certified Smart Cards: Coppersmith in the Wild. Advances in Cryptology - ASIACRYPT 2013 Lecture Notes in Computer Science, 341–360. doi: 10.1007/978-3-642-42045-0_18

Nemec, M., Sys, M., Svenda, P., Klinec, D., Matyas, V. (2017). The Return of Coppersmiths Attack. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. doi: 10.1145/3133956.3133969

LLL algorithm and usage in cryptography

Ariana

libgen/scihub

May 16, 2020