Decodificando reticulados para criptografia

Ramon Ribeiro

Instituto de Computação - UNICAMP

25/02/2018

Introdução

Algoritmo de Shor

Em 1997, Peter Shor descreveu um algoritmo capaz de resolver ambos problemas de fatoração de inteiros e logaritmos discretos em tempo polinomial, para ser executado em computadores quânticos.

Sistemas pós-quânticos

São sistemas criptográficos baseados em problemas para os quais não se conhece algoritmos (quânticos ou não) que resolvam seu problema base em tempo polinomial.
Alguns deles são baseados em problemas difíceis em reticulados.

Tópicos

- Introdução a criptografia
- Principais problemas em reticulados para criptografia
- Principais reticulados para criptografia
- Escolha do reticulado que sera estudado
- Decodificador e sistema criptográfico que conversam entre si
- Comparação dos dados gerados

Índice

Empacotamento

Se denota como δ_n a maior densidade de empacotamento por esferas em R^n.
Sendo que um reticulado $\mathcal{L}_n$ é um subgrupo de R^n, um reticulado perfeito seria aquele cuja densidade δ fosse igual a δ_n. Implicando em um base melhor para criptografia, visto que possuem as menores distancias possíveis entre vetores.

Reticulados Perfeitos

dim.	Nr. of perfect lattices	Absolute maximum of γ realized by
2	1 (Lagrange)	A_{hex}
3	1 (Gauss)	A_3
4	2 (Korkine & Zolotareff)	D_4
5	3 (Korkine & Zolotareff)	D_5
6	7 (Barnes)	E_6
7	33 (Jaquet)	E_7
8	10916 (Dutour, Schürmann & Vallentin)	E_8

Reticulados com os Melhores Empacotamentos Conhecidos

Dim.	Simbolo	Nome
9	Λ_9	Laminated lattice
10	K_{10}	Coxeter Todd lattice
11	Λ_{11}	Laminated lattice
12	K_{12}	Coxeter Todd lattice
16	BW_{16}	Barnes-Wall lattice
24	Λ_{24}	Leech lattice

Reticulados Criptograficamente Relevantes

Dim.	Simbolo	Nome
8	E_8	Perfect 8 dim. lattice
9	Λ_9	Laminated lattice
16	BW_{16}	Barnes-Wall lattice
24	Λ_{24}	Leech lattice

CVP e SVP

- The Shortest Vector Problem (SVP) : Encontre o menor $v \in \mathfrak{L}$.
- The Closest Vector Problem (CVP) : Dado $w \in \mathbb{R}^n \wedge w \notin \mathfrak{L}$, encontre $v \in \mathfrak{L}$ tal que seja mínima $||w - v||$
- apprSVP : Encontre $v \in \mathfrak{L}$ tal que $||v|| \leq \psi(n)||V_{shortest}||$
- apprCVP : Dado $w \in \mathbb{R}^n \wedge w \notin \mathfrak{L}$, encontre $v \in \mathfrak{L}$ tal que $||w - v|| \leq \psi(n)||V_{closest}||$

Dificuldade desses problemas

- Computacionalmente, tanto o SVP quanto o CVP se tornam tão mais difíceis quanto maior a dimensão do reticulado em análise.
- O CVP é considerado NP-difícil.
- O SVP é considerado NP-difícil sobre certas hipóteses de redução aleatórias

O Problema LWE

Para n, q inteiros positivos, χ uma distribuição de probabilidade em $\mathbb{Z}$ e s um vetor secreto em $\mathbb{Z}_q^n$.
Denotamos, portanto, $L_{s,\chi}$ como a distribuição de probabilidade em $\mathbb{Z}_q^n x \mathbb{Z}_q$, ao escolher $a \epsilon \mathbb{Z}_q^n$ aleatória e uniformemente, escolhendo $e \epsilon \mathbb{Z}$ de acordo com χ e considerando-o contido em $\mathbb{Z}_q$.

Obtendo:

$$(a, c) = (a, \langle a, s \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$$

Decision-LWE e Search-LWE

Decision-LWE

É o problema em decidir se os pares $(a, c) \in \mathbb{Z}_q^n x \mathbb{Z}_q$ são obtidos de acordo com $L_{s,\mathcal{X}}$ ou a distribuição uniforme em $\mathbb{Z}_q^n x \mathbb{Z}_q$

Decision-LWE e Search-LWE

Decision-LWE

É o problema em decidir se os pares $(a, c) \in \mathbb{Z}_q^n x \mathbb{Z}_q$ são obtidos de acordo com $L_{s,\mathcal{X}}$ ou a distribuição uniforme em $\mathbb{Z}_q^n x \mathbb{Z}_q$

Search-LWE

É o problema em recuperar s de $(a, c) = (a, \langle a, s \rangle + e) \in \mathbb{Z}_q^n x \mathbb{Z}_q$, obtido de $L_{s,\mathcal{X}}$

Ring-LWE

Seja K um conjunto numerico, θ_K é o seu anel ciclotômico de inteiros e $q > 2$ é um inteiro racional.

Search-RLWE

É o problema em recuperar um segredo $s \in \theta_K^V / q\theta_K^V$, com θ_K^V sendo o dual de θ_K, de varias amostras arbitrarias $(a_i, a_i.s + e_i)$. Em que cada a_i é uniformemente amostrada de $\theta_K / q\theta_K$ e cada e_i é um pequeno elemento aleatorio de $K_{\mathbb{R}} := K \otimes_{\mathbb{Q}} \mathbb{R}$.

Ring-LWE

Seja K um conjunto numerico, θ_K é o seu anel de inteiros e $q > 2$ é um inteiro racional.

Search-RLWE

É o problema em recuperar um segredo $s \in \theta_K^V / q\theta_K^V$, com θ_K^V sendo o dual de θ_K, de varias amostras arbitrarias $(a_i, a_i.s + e_i)$. Em que cada a_i é uniformemente amostrada de $\theta_K / q\theta_K$ e cada e_i é um pequeno elemento aleatorio de $K_{\mathbb{R}} := K \otimes_{\mathbb{Q}} \mathbb{R}$.

Decision-RLWE

É o problema em distinguir entre vários pares arbitrários (a,c), aqueles com um s escolhido uniformemente comum entre eles.

SIS - Short Interger Solutions (Decision-LWE)

Pode-se tentar encontrar um vetor pequeno v tal que v.a=0, a fim de distinguir se m casos (a,c) seguem/pertencem a $L_{s,\mathcal{X}}$, e, portanto, satisfazem $c = \langle as \rangle + e$, ou se c é uniformemente aleatório.

SIS - Perspectiva de reticulados

Dessa perspectiva, procura-se obter um vetor v no reticulado dual escalado (por q) gerado por a.
Considere $\langle v, c \rangle$, se $c = as + e$ então $\langle v, c \rangle$ seguindo a distribuição gaussiana em $\mathbb{Z}$ (mod. q). Do outro lado, se c for uniforme, então $\langle v, c \rangle$ é uniforme em $\mathbb{Z}_q$.
Deve-se manter $||v||$ pequeno o suficiente para manter a distribuição gaussiana de $\langle v, e \rangle$ boa para se distinguir de distribuições aleatórias.

BDD - Bounded Distance Decoder (Search-LWE)

Dadas amostras de $(a, c) = (a, as + e)$ dentro de $L_{s,\chi}$, pode-se observar que c é próximo a uma combinação linear das colunas de a. Além disso, o ruido é gaussiano, sendo que quase todo ele esta contido em três vezes o desvio padrão $(\frac{3\alpha q}{\sqrt{2\pi}})$.
Logo, o problema seria achar o ponto $w = as$ do qual c esta contido através do limitante. A partir disso recuperaremos s com álgebra linear.

Solving for s (Search-LWE)

Usa uma estrategia similar a anterior, só que busca s diretamente, tal que $||as - c||$ seja mínimo.

BBD - Tipos de algoritmos (Search-LWE)

- Parallel Bounded Distance Decoding Algorithm - Micciancio e Nicolosi
- List-Decoding Algorithm - Grigorescu e Peikert†

Reticulado de Barnes-Wall

Sendo $\mathbb{G} = \mathbb{Z} + i\mathbb{Z}$ o grupo de inteiros gaussianos, definimos $\phi = 1 + i$ como o inteiro gaussiano de menor norma.
Dessas definições, escreve-se o reticulado assim:

$$BW^n = \begin{bmatrix} 1 & 1 \\ 0 & \phi \end{bmatrix}^{\otimes n}.$$

Definindo $N = 2^n$ como a dimensão do reticulado de Barnes-Wall, vê-se que ele possui distancia miníma $d_{min}(BW^n) = \sqrt{N}$, volume $V(BW^n) = \sqrt{N^N}$, e ganho nominal de código $\gamma_c(BW^n) = \sqrt{N}$.

Reticulado de Barnes-Wall

Definindo o produto de Kronecker deste modo:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}$$

Temos que o reticulado pode ser representado assim:

$$BW^n = \begin{bmatrix} BW^{n-1} & BW^{n-1} \\ \mathbf{O} & \phi \cdot BW^{n-1} \end{bmatrix}$$

Parallel Bounded Distance Decoding Algorithm

O tempo de uso do algorítimo (paralelo), medido em termos de operações aritméticas é $O(Nlog^2N/\sqrt{p})$.

Parallel Bounded Distance Decoding Algorithm

Algorithm 1 Parallel Bounded Distance Decoder (BDD) for Barnes-Wall Lattices

function $\textsc{ParBW}(p, \mathbf{s})$
  if $p < 4$ or $\mathbf{s} \in \mathbb{C}^1$ **then**
    return $\textsc{SeqBW}(0, \mathbf{s})$ ▷ Run the sequential decoder from Section 3
  else
    $[\mathbf{s}_0, \mathbf{s}_1] \leftarrow \mathbf{s}$ ▷ Split $\mathbf{s}$ into two halves
    $[\mathbf{s}_-, \mathbf{s}_+] = (\phi/2) \cdot [\mathbf{s}_0 - \mathbf{s}_1, \mathbf{s}_0 + \mathbf{s}_1]$ ▷ Compute $T(\mathbf{s})$
    $\begin{bmatrix} \mathbf{z}_0 \\ \mathbf{z}_1 \\ \mathbf{z}_- \\ \mathbf{z}_+ \end{bmatrix} \leftarrow \begin{bmatrix} \textsc{ParBW}(p/4, \mathbf{s}_0) \\ \textsc{ParBW}(p/4, \mathbf{s}_1) \\ \textsc{ParBW}(p/4, \mathbf{s}_-) \\ \textsc{ParBW}(p/4, \mathbf{s}_+) \end{bmatrix}$ ▷ Execute recursive calls in parallel
    $\mathbf{z}_0^- \leftarrow [\mathbf{z}_0, \mathbf{z}_0 - 2\phi^{-1}\mathbf{z}_-]$ ▷ Compute 4 candidate vectors
    $\mathbf{z}_0^+ \leftarrow [\mathbf{z}_0, 2\phi^{-1}\mathbf{z}_+ - \mathbf{z}_0]$
    $\mathbf{z}_1^- \leftarrow [2\phi^{-1}\mathbf{z}_- + \mathbf{z}_1, \mathbf{z}_1]$
    $\mathbf{z}_1^+ \leftarrow [(2\phi^{-1}\mathbf{z}_+ - \mathbf{z}_1, \mathbf{z}_1]$
    $\mathbf{z} = \underset{\mathbf{z}' \in \{\mathbf{z}_0^-, \mathbf{z}_0^+, \mathbf{z}_1^-, \mathbf{z}_1^+\}}{\operatorname{argmin}} \{\|\mathbf{s} - \mathbf{z}'\|\}$ ▷ Select the candidate closest to $\mathbf{s}$
    return $\mathbf{z}$
  end if
end function

Parallel BDD - Sequential BDD

Algorithm 2 Sequential Bounded Distance Decoder for Barnes-Wall Lattices and Their Principal Sublattices

```
function SeqBW(r, s)
    if s ∈ ℂ^N with N ≤ 2^r then
        return ⌈s⌋ ∈ 𝔾^N            ▷ Round s component-wise to the closest Gaussian integer
    else
        b ← ⌈ℜ(s)⌋ + ⌈ℑ(s)⌋ mod 2                 ▷ Compute binary target component-wise
        ρ = 1 − 2 max(|ℜ(s) − ⌈ℜ(s)⌋|, |ℑ(s) − ⌈ℑ(s)⌋|)   ▷ Compute the reliability information
        t ← (b, ρ)                   ▷ Component-wise pairing, i.e., t_j = (b_j, ρ_j)
        ψ(c) ← RMDec^ψ(r, t)         ▷ Call the Reed-Muller soft-decision decoder
        v ← SeqBW(r + 1, (s − ψ(c))/φ)
        return ψ(c) + φ v
    end if
end function
```

Parallel BDD - Definições

Cada vetor do reticulado BW^n pode ser unicamente representado assim:

$$v = \sum_{r=0}^{n-1} \phi^r \psi(c_r) + \phi^n c_n$$

em que $c_n \in \mathbb{G}^N$ e $c_r \in RM_r^n$, para r=0,1,...,n-1.
Também tem-se que:

$$\begin{cases} \psi(0) = 0 \\ \psi(1) = 1 \\ \psi([\mathbf{u}, \mathbf{u} \oplus \mathbf{v}]) = [\psi(\mathbf{u}), \psi(\mathbf{u}) + \psi(\mathbf{v})] \end{cases}$$

Parallel BBD - Soft Decision

Algorithm 3 Soft Decision Decoder for Reed-Muller Codes

function $\text{RMDEC}^{\psi}(r, \mathbf{t})$ ▷ Input: $r \geq 0$, $\mathbf{t} \in (\{0,1\} \times [0,1])^N$
  if $r = 0$ **then**
    if $\sum_{b_j=0} \rho_j > \sum_{b_j=1} \rho_j$ **then**
      return $[0, \ldots, 0]$
    else
      return $[1, \ldots, 1]$
    end if
  else if $N = 2^r$ **then**
    return $[b_1, \ldots, b_N]$ ▷ where $(b_j, \rho_j) = t_j$
  else
    $[\mathbf{t}^0, \mathbf{t}^1] \leftarrow \mathbf{t}$ ▷ Split $\mathbf{t}$ into halves
    for $j = 1, \ldots, N/2$ **do**
      $t_j^+ \leftarrow (b_j^0 \oplus b_j^1, \min(\rho_j^0, \rho_j^1))$ ▷ where $(b_j^0, \rho_j^0) = t_j^0$ and $(b_j^1, \rho_j^1) = t_j^1$
    end for
    $\mathbf{v} \leftarrow \text{RMDEC}^{\psi}(r-1, \mathbf{t}^+)$
    for $j = 1, \ldots, n/2$ **do**
      if $b_j^0 \oplus b_j^1 = v_j \bmod 2$ **then**
        $t_j^- \leftarrow (b_j^0, (\rho_j^0 + \rho_j^1)/2)$
      else
        $t_j^- = (b_j^0 \oplus \text{EVAL}(\rho_j^0 < \rho_j^1), |\rho_j^0 - \rho_j^1|/2)$ ▷ where $\text{EVAL}(\varphi) = 1$ iff formula φ holds
      end if
    end for
    $\mathbf{u} \leftarrow \text{RMDEC}^{\psi}(r, \mathbf{t}^-)$
    return $[\mathbf{u}, \mathbf{u} + \mathbf{v}]$.
  end if
end function

Referências

- Sphere packings and lattice sphere packings - Mathieu Dutour Sikiric
- A Simple Construction for the Barnes-Wall Lattices
- Efficient Bounded Distance Decoders for Barnes-Wall Lattices - Daniele Micciancio and Antonio Nicolosi April 30, 2008
- List Decoding Barnes-Wall Lattices - Elena Grigorescu and Chris Peikert April 10, 2012

Referências

- On the concrete hardness of Learning with Errors - Martin R. Albrecht, Rachel Player, and Sam Scott
- On Lattices, Learning with Errors, Random Linear Codes, and Cryptography - Oded Regev May 2, 2009
- Post-Quantum Cryptography - Daniel J. Bernstein and Johannes Buchmann (Lattice-based Cryptography Chapter)
- Lecture 3 - CVP Algorithm - Lecturer: Oded Regev