UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP

Criptografia baseada em Reticulados

Maiara Francine Bollauf

21 de Novembro de 2014

1 Reticulados

1.1 Problemas envolvendo reticulados

Vamos apresentar agora, os problemas existentes no estudo de reticulados, a dizer: o problema do vetor mais curto (SVP - *shortest vector problem*) e o problema do vetor mais próximo (CVP - *closest vector problem*), usando como referência os textos de 14 e 10.

As construções em criptografia baseada em reticulados são elaboradas usando os problemas que são considerados difíceis em reticulados, são eles:

PROBLEMA DO VETOR MAIS CURTO (SVP): Consiste em encontrar o vetor não nulo mais curto em um reticulado $\mathscr{L}$, isto é, encontrar um vetor não nulo $v \in \mathscr{L}$ que minimiza a norma euclidiana $||v||$.

PROBLEMA DO VETOR MAIS PRÓXIMO (CVP): Dado um vetor $w \in \mathbb{R}^n$, mas não está no reticulado $\mathscr{L}$, busca-se encontrar o vetor $v \in \mathscr{L}$ que é mais próximo a w, isto é, encontrar um vetor $v \in \mathscr{L}$ que minimiza a norma euclidiana $||w - v||$.

Observação 1.1. Note que pode existir mais de um vetor não nulo que satisfaça o SVP em um reticulado. Por exemplo,em $\mathbb{Z}_2$, todos os vetores na forma $(0, \pm 1)$ e $(\pm 1, 0)$ são soluções para o SVP. Esse é o motivo pelo qual o problema SVP exige "um" vetor mais curto e não "o" vetor mais curto. Uma observação similar se aplica ao CVP.

Ambos SVP e CVP são problemas elaborados e ambos se tornam computacionalmente difíceis a medida que a dimensão n do reticulado cresce. Por outro lado, até as soluções aproximadas para os problemas SVP e CVP possuem suas aplicações em diferentes campos da matemática pura e aplicada. Sabemos ainda que o problema CVP é $NP-$completo, como provado em 5, e o problema SVP é $NP-$completo sob certas "reduções de hipóteses", o que pode ser melhor explicado em 2.

Observação 1.2. Em geral, ambos SVP e CVP são considerados como sendo problemas difícies em reticulados, mas na prática, é complicado atingir a "generalidade completa" tão idealizada, de acordo com 10. Em um cenário real, criptossistemas baseados em problemas

$NP-$completos tendem a recair em uma subclasse particular de problemas, seja para alcançar eficiência ou permitir a criação de um alçapão [1]. Quando isso acontece, existe sempre a possibilidade de alguma propriedade especial da subclasse de problemas tornar o problema inicial mais fácil de ser resolvido no caso geral.

Vamos descrever algumas variações dos problemas SVP e CVP que são aplicados tanto na teoria quanto na prática.

PROBLEMA DA BASE MAIS CURTA (SBP): Consiste em encontrar uma base $v_1, \ldots, v_n$ para um reticulado tal que ela seja a mais curta em algum parâmetro. Por exemplo, podemos quer que

$$\max_{1 \leq i \leq n} ||v_i|| \text{ ou } \sum_{i=1}^{n} ||v_i||^2$$

sejam mínimos. Existem diferentes versões desse problema, dependendo dos parâmetros usados para medir o tamanho da base.

PROBLEMA DO VETOR MAIS CURTO APROXIMADO (APPRSVP): Seja $\psi(n)$ uma função de n. Em um reticulado $\mathscr{L}$ de dimensão n, buscamos encontrar um vetor não nulo tal que não seja maior do que $\psi(n)$ vezes maior do que qualquer outro vetor não nulo mais curto. Em outras palavras, se v_s é o vetor mais curto em $\mathscr{L}$, queremos encontrar um vetor não nulo $v \in \mathscr{L}$ que satisfaz:

$$||v|| \leq \psi(n)||v_s||$$

E cada escolha distinta para $\psi(n)$ define um problema APPRSVP distinto.
Por exemplo, considere um algoritmo que queira encontrar $v \in \mathscr{L}$ satisfazendo

$$||v|| \leq 3\sqrt{n}||v_s|| \text{ ou } ||v|| \leq 2^{n/2}||v_s||.$$

Claramente um algoritmo que resolve o primeiro dos problemas é mais forte do que um algoritmo que resolve o segundo.

PROBLEMA DO VETOR MAIS PRÓXIMO APROXIMADO (APPRCVP): É o igual ao APPRSVP, mas agora estamos procurando um vetor que aproxime a solução do CVP, ao invés de aproximar a solução do SVP.

Os resultados vistos nas aulas anteriores definem cotas superiores e inferiores para reticulados em temos de $\det(\mathscr{L})$ e $dim(\mathscr{C})$ para o vetor mais curto em $\mathscr{L}$. Isso pode ser observado, por exemplo, em:

Teorema 1.3. 10 **(Teorema de Minkowski)** Seja $\mathscr{L} \subset \mathbb{R}^n$ um reticulado de dimensão n e seja $S \subset \mathbb{R}^n$ um conjunto convexo e simétrico cujo volume satisfaz:

$$vol(S) > 2^n \det(\mathscr{L}).$$

[1] 17 Seja $f : \mathscr{D} \longrightarrow \mathscr{I}$ uma função. Tal função é chamada de alçapão se as seguintes propriedades forem atendidas:

- **Unidirecional:** f é uma função unidirecional, ou seja, se dado $x \in \mathscr{D}$ é possível calcular $y = f(x)$ em tempo polinomial e se dado $y \in \mathscr{I}$ não se conhece algoritmo em tempo polinomial que calcule $x \in \mathscr{D}$ tal que $f(x) = y$.
- **Informação secreta:** Existe uma informação secreta κ tal que dado $y \in \mathscr{I}$ e κ é possível calcular em tempo polinomial $x \in \mathscr{D}$ tal que $f(x) = y$.

Então, S contém um vetor não nulo do reticulado $\mathscr{L}$.

Para relembrar o próximo resultado, vamos definir:

Definição 1.4. Dizemos que K é um **corpo convexo** se K for um conjunto convexo limitado e tal que $\text{int}(S) \neq \emptyset$.

Definição 1.5. Seja K um corpo convexo com $a \in \text{int}(K)$. Então, para cada $x \in \mathbb{R}^n$, existem números positivos λ tais que $x \in \lambda K$. Definimos então uma função F_K da seguinte maneira:

$$F_K(x) = inf\{\lambda : \lambda > 0, x \in \lambda K\}.$$

Tal função é chamada de **função distância** ou **função de Gauge**.

Definição 1.6. Definimos o **primeiro mínimo de** $\mathscr{L}$ com relação à um corpo convexo K como

$$\lambda_{1,k}(\mathscr{L}) = \min_{x \in \mathscr{L} \setminus \{0\}} F_K(x).$$

Em criptografia, o valor do primeiro mínimo $\lambda_{1,k}$ é importante para estimar parâmetros para reticulados que não são munidos de alguma estrutura especial e decidir se o algoritmo baseado em tal reticulado terá sucesso em sua implementação. Experimentos vem sendo desenvolvidos por Gama, Nguyen e Regev (7) e mostram que é suficiente, para reticulados aleatórios, estimar o valor do primeiro mínimo $\lambda_{1,k}$, sem precisar resolver o problema SVP, por exemplo.

O teorema que segue nos dá uma estimativa para o primeiro mínimo de um reticulado $\mathscr{L}$ em corpos convexos.

Teorema 1.7. 6 Seja $K \subset \mathbb{R}^n$ um corpo convexo e $\mathscr{L}$ um reticulado. Então,

$$\lambda_{1,k} \leq 2\left(\frac{\det(\mathscr{L})}{vol(K)}\right)^{1/n},$$

onde $\lambda_{1,k}$ é conforme na Definição 1.6.

Teorema 1.8. 6 Seja $\mathscr{L}$ um reticulado. Existe $x \in \mathscr{L}$ tal que

$$||x||_p \leq n^{1/p}(\det(\mathscr{L})^{1/n}.$$

Se tomarmos $p = 2$, temos então o seguinte resultado:

Teorema 1.9. 10 Todo reticulado $\mathscr{L}$ de dimensão n contém um vetor não nulo $x \in \mathscr{L}$ que satisfaz:

$$||x|| \leq \sqrt{n}\det(\mathscr{L})^{1/n}.$$

Note que no caso em que $p = 2$ temos uma aproximação ainda melhor para a cota estabelecida pelo Teorema 1.9. De fato, temos que o corpo convexo K do Teorema 1.7 é tal que $K = B_2(r)$, ou seja, é a bola com a norma 2 de raio r e vale que:

$$vol(K) = \frac{\pi^{n/2}}{\left(\frac{n}{2}\right)!}.$$

Com isso, $vol(K)^{1/n} = \dfrac{\sqrt{\pi}}{\left(\left(\frac{n}{2}\right)!\right)^{1/n}}$. Utilizando a aproximação de Stirling, temos que:

$$\left(\frac{n}{2}\right)! \approx (\sqrt{2\pi e})e^{-n/2}\left(\frac{n}{2}\right)^{\frac{n}{2}-\frac{1}{2}}$$

$$\left(\left(\frac{n}{2}\right)!\right)^{1/n} \approx (\sqrt{2\pi e})^{1/n} e^{-1/2} \left(\frac{n}{2}\right)^{\frac{1}{2}-\frac{1}{2n}}$$

Se n é assintótico:

$$\left(\left(\frac{n}{2}\right)!\right)^{1/n} \approx \left(\frac{n}{2e}\right)^{1/2}$$

Então, temos o seguinte resultado:

$$(vol(K))^{1/n} \approx \frac{\sqrt{2\pi}}{\sqrt{n}} e \leq \frac{2\sqrt{n}}{\sqrt{2\pi e}} (\det(\mathscr{L}))^{1/n}$$

e a constante $\frac{2}{\sqrt{2\pi e}} \simeq 0.4839$ é uma excelente aproximação, neste caso, para a cota estabelecida para o Teorema 1.9.

Definição 1.10. Para uma dada dimensão n, a constante de Hermite γ_n é definida como sendo o menor valor tal que todo reticulado $\mathscr{L}$ de dimensão n contém um vetor não nulo $v \in \mathscr{L}$ que satisfaz:

$$||v||_2 \leq \gamma_n \det(\mathscr{L})^{2/n}.$$

Observe que na versão apresentada do Teorema 1.9, temos que $\gamma_n \leq n$. O valor exato de γ_n, de acordo com 10, é conhecido somente para $1 \leq n \leq 8$ e para $n = 24$:

$$\gamma_2^2 = \frac{4}{3},\ \gamma_3^3 = 2,\ \gamma_4^4 = 4,\ \gamma_5^5 = 8,\ \gamma_6^6 = \frac{64}{3},\ \gamma_7^7 = 64,\ \gamma_8^8 = 256 \text{ e } \gamma_{24} = 4.$$

É importante ressaltar que encontrar o valor exato para a constante de Hermite γ_n é um problema muito difícil e foi um dos principais tópicos do estudo de Geometria dos Números por Minkowski em 15. Para as propostas criptográficas, o valor de γ_n é interessante quando n for muito grande ou assintótico. Para valores de n consideravelmente grandes, sabe-se que a constante de Hermite satisfaz:

$$\frac{n}{2\pi e} \leq \gamma_n \leq \frac{n}{\pi e},$$

onde $\pi = 3.14159...$ e $e = 2.71828...$ são as constantes conhecidas.

1.2 Algoritmo de Babai

Se um reticulado $\mathscr{L}$ possui uma base $v_1, v_2, \ldots, v_n$ formada por vetores dois a dois ortogonais, ou seja,

$$v_i \cdot v_j = 0,\ \forall i \neq j$$

então é fácil resolver os problemas SVP e CVP. Nesta seção vamos, então, compreender o algoritmo desenvolvido por Babai 3 em 1985 para abordar esses problemas difíceis em reticulados considerando tais hipóteses. Como referência, usamos o texto de 10.

Para resolver SVP, observe que o comprimento de qualquer vetor no reticulado $\mathscr{L}$ é dado pela fórmula:

$$||a_1 v_1 + a_2 v_2 + \cdots + a_n v_n||^2 = {a_1}^2 ||v_1||^2 + {a_2}^2 ||v_2||^2 + \cdots + {a_n}^2 ||v_n||^2.$$

Como $a_1, \ldots, a_n \in \mathbb{Z}$, temos que os vetores mais curtos não nulos em $\mathscr{L}$ são simplesmente os vetores mais curtos no conjunto $\{\pm v_1, \pm v_2, \ldots, \pm v_n\}$.

De modo similar, suponha que queremos encontrar o vetor em $\mathscr{L}$ que está mais próximo a um dado vetor $w \in \mathbb{R}^n$. Escrevemos, primeiramente:

$$w = t_1 v_1 + t_2 v_2 + \cdots + t_n v_n, \text{ com } t_1, \ldots, t_n \in \mathbb{R}.$$

Então, para $v = a_1 v_1 + a_2 v_2 + \cdots + a_n v_n \in \mathscr{L}$, temos:

$$||v - w|| = (a_1 - t_1)^2 ||v_1||^2 + (a_2 - t_2)^2 ||v_2||^2 + \cdots + (a_n - t_n)^2 ||v_n||^2. \quad (1)$$

Como a_i deve ser inteiro, a Equação 1 será minimizada se tomarmos cada a_i como sendo o inteiro mais próximo ao correspondente t_i.

Observando esse processo, é difícil não querer utilizá-lo com uma base qualquer não ortogonal, porém, se a base não for ortogonal, o algoritmo não irá funcionar corretamente, como veremos a seguir.

A base $\{v_1, v_2, \ldots, v_n\}$ para o reticulado $\mathscr{L}$ determina uma região fundamental. Sabemos que as translações de $\mathscr{F}$ por elementos de $\mathscr{L}$ preenchem todo o espaço do $\mathbb{R}^n$, logo, cada $w \in \mathbb{R}^n$ está em uma única translação $\mathscr{F} + v$ de $\mathscr{F}$ por um elemento $v \in \mathscr{L}$. Consideramos o vértice do paralelogramo $\mathscr{L} + v$ que está mais próximo a w como sendo nossa solução hipotética para o problema CVP. É fácil encontrar o vértice mais próximo, já que:

$$w = v + \varepsilon_1 v_1 + \varepsilon_2 v_2 + \cdots + \varepsilon_n v_n \text{ para algum } 0 \leq \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n < 1,$$

e então, somente substituímos ε_i por 0 se ele for menor do que $\frac{1}{2}$ e substituímos por 1 se ele for maior ou igual a $\frac{1}{2}$.

Vamos introduzir uma noção ainda não mencionada envolvendo reticulados, que são as bases "boas" e as bases "ruins". Uma base boa para um reticulado, é uma base que consiste em vetores curtos e ortogonais entre si e uma base ruim para um reticulado, por sua vez, é a formada por vetores longos que geralmente apontam para a mesma direção (ou oposta) ou ainda por vetores que possuem o ângulo entre eles muito pequenos.

Se tentarmos resolver o problema CVP com uma base ruim, enfrentaremos problemas ao encontrar o vértice mais próximo à um dado vetor que se encontra dentro do paralelogramo definido por essa base. E é importante ressaltar que os problemas começam a ficar muito piores a medida que a dimensão do reticulado vai aumentando.

O método geral para encontrar a solução do problema CVP, chamado de Algoritmo de Babai para o Vértice mais Próximo, está explicitado no Algoritmo 1.

Algoritmo 1 Algoritmo de Babai para o Vértice mais Próximo

Seja $\mathscr{L} \subset \mathbb{R}^n$ um reticulado com base $v_1, v_2, \ldots, v_n$ e seja $w \in \mathbb{R}^n$ um vetor arbitrário. Se os vetores da base forem suficientemente ortogonais um ao outro, então resolvemos o CVP da seguinte maneira:
(1) Escreve $w = t_1 v_1 + t_2 v_2 + \cdots + t_n v_n$ com $t_1, \ldots, t_n \in \mathbb{R}$.
(2) Defina $a_i = \lfloor t_i \rceil$ para $i = 1, 2, \ldots, n$.
(3) Devolva o vetor $v = a_1 v_1 + a_2 v_2 + \cdots + a_n v_n$.

Em geral, se os vetores da base são razoavelmente ortogonais uns aos outros, o algoritmo resolve algumas versões do APPRCVP, mas se os vetores da base forem altamente não ortogonais, então o vetor devolvido pelo algoritmo estará muito longe do vetor mais próximo ao vetor w.

2 Criptografia baseada em reticulados

A criptografia baseada em reticulados consiste, atualmente, em uma das mais promissoras alternativas para a evolução da criptografia clássica e seu estudo iniciou-se com a publicação de Ajtai, em 1996, intitulada "*Generating hard instances of Lattice problems*", em 1996.

Existiram duas grandes motivações para a introdução desses criptossistemas: o interesse em construir sistemas criptográficos baseados em vários problemas difíceis em matemática e a crença de que criptossistemas baseados em reticulados podem ser mais rápidos que fatorações ou sistemas baseados em logaritmos discretos , como o RSA, por exemplo.

2.1 Criptossistem GGH

Os problemas SVP e CVP, considerados difíceis em um reticulado de dimensão n, quando n é consideravelmente grande, serviram de base para muitos criptossistemas introduzidos em meados dos anos 90. Os mais importantes deles foram: o criptossistema de Ajtai-Dwork 1, o criptossistema GGH atribuído a Goldreich, Goldwasser and Halevi 8 e o criptossistema NTRU proposto por Hoffstein, Pipher, and Silverman 11.

O sistema de Ajtai-Dwork é particularmente interessante, de acordo com 10, pois os autores mostraram que o criptosssitema desenvolvido é provavelmente seguro a não ser que o pior caso do problema em reticulados possa ser resolvido por um algoritmo em tempo polinomial. Em compensação, esse importante resultado teórico ainda nos diz que o tamanho das chaves é aproximado por $\mathcal{O}(n^4)$, o que implica em chaves enormes.

De modo informal, o sistema GGH se explica da seguinte forma: a chave privada de Alice é uma base boa β_{boa} para um reticulado $\mathscr{L}$ e sua chave pública é uma base ruim β_{ruim} para o mesmo reticulado $\mathscr{L}$. A mensagem de Bob é um vetor binário m, que ele utiliza para formar uma combinação linear $\sum m_i v_i$ dos vetores $v_i \in \beta_{\text{ruim}}$. Bob então perturba essa soma adicionando à ela um pequeno vetor r aleatório. O resultado é um vetor w que difere de um vetor v do reticulado por um vetor r. Como Alice conhece uma base boa para $\mathscr{L}$, ela pode utilizar o algoritmo de Babai para encontrar v e expressá-lo em termos da base ruim β_{ruim} e recuperar m. Se um atacante, Tom, conhece somente a base β_{ruim}, ele é incapaz de resolver o problema CVP em $\mathscr{L}$.

Munidos de tais informações, podemos esquematizar um algoritmo que explique o funcionamento do criptossistema GGH, conforme apresentado no Algoritmo 4.1

Algoritmo 2 O sistema criptográfico de chave pública GGH

Escolha um reticulado $\mathscr{L}$.
Chave privada: $\{v_1, \ldots, v_n\}$ como sendo uma base boa e curta para o reticulado $\mathscr{L}$.
Chave pública: $\{w_1, \ldots, w_n\}$ como sendo uma base ruim e longa para o reticulado $\mathscr{L}$.
Encriptação de uma mensagem $m \in \mathbb{F}_2{}^n$: Escolha um vetor de perturbação r. A mensagem cifrada, por sua vez, é dada por: $e = m_1 w_1 + m_2 w_2 + \cdots + m_n w_n + r$. Note que a mensagem cifrada $e \notin \mathscr{L}$.
Decriptação: Encontre um vetor $u \in \mathscr{L}$ que é próximo a e. Se r for pequeno o suficiente, então $u = m_1 w_1 + m_2 w_2 + \cdots + m_n w_n$, com isso, basta resolver o CVP para e em $\mathscr{L}$ para recuperar m. A chave privada, que é a base boa para o reticulado $\mathscr{L}$, pode ser usada para encontrar u. Escreva, primeiramente, $e = \mu_1 v_1 + \mu_2 v_2 + \cdots + \mu_n v_n$, com $\mu_1, \mu_2, \ldots, \mu_n \in \mathbb{R}$. A aproximação $\mu_1, \mu_2, \ldots, \mu_n$ para o menor inteiro: $\lfloor \mu_1 \rceil v_1 + \cdots + \lfloor \mu_n \rceil v_n$ será igual a u.

Como pudemos notar, o criptossistema GGH é dos sistemas mais intuitivos baseados em reticulados e a sua segurança reside na dificuldade em resolver o CVP usando uma base altamente não ortogonal. O criptossistema GGH foi submetido a ataques criptoanalíticos 16 com parâmetros de segurança relativamente grandes e pode ser considerado inseguro de um ponto de vista prático, de acordo com 14, o que é uma grande desvantagem em sua aplicação.

Uma outra forma de atacar o criptossistema GGH é tentar melhor a base ruim, que é a chave pública, afim de tornar seus vetores menores e mais ortogonais. Em dimensão 2 esse problema pode ser facilmente resolvido, segundo 4, já em dimensões maiores, esse problema é considerado difícil.

Além disso, vimos que a chave pública do sistema GGH é uma base de um reticulado $\mathscr{L}$, assim, seu tamanho é de aproximadamente $\mathscr{O}(n^2)$ bits, que muito grande, tornando o uso desse criptossistema ineficiente em termos práticos.

2.2 O criptossistema NTRU

O criptossistema NTRU é conhecido como sendo um dos mais práticos e promissores algoritmos baseados em reticulado. Sua segurança está baseada, assim como o GGH, na dificuldade em resolver o problema CVP em reticulados. Esse sistema foi desenvolvido por Hoffstein, Pipher e Silverman 11 entre os anos de 1994 e 1996 e tornou-se público em um evento de criptografia em 1996.

Criptossistemas baseados nas dificuldades sugeridas pela fatoração inteira ou no problema do logaritmo discreto são baseadas em teoria de grupos, pois seus problemas difíceis requerem somente uma operação. Para RSA consideramos, por exemplo, o grupo das unidades módulo m para algum módulo m que pode ser primo ou composto e o grupo da multiplicação módulo m. Já para os esquemas baseados em curvas elípticas, o grupo em questão é o grupo dos pontos de uma curva elíptica módulo p e o grupo da adição de curvas elípticas.

Quando necessitamos de mais de uma operação, pensamos em anéis, já que um anel é uma estrutura algébrica munido das operações de soma e multiplicação, além da lei da distributividade. O sistema NTRU é baseado originalmente em anéis, mas pode ser descrito de maneira equivalente usando reticulados, que é o que faremos nessa seção, usando como referência os textos de 10, 4 e 14.

A proposta original do sistema NTRU, por sua vez, foi desenvolvida usando a estrutura de anéis convolucionais de polinômios, ou seja, o anel R definido por:

$$R = \frac{\mathbb{Z}_q[x]}{(x^N - 1)},$$

onde os elementos desse anel são polinômios da forma:

$$a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{N-1} x^{(n-1}.$$

Definição 2.1. A multiplicação no anel R é dada pela operação de convolução:

$$\left(\sum_{i=0}^{N-1} a_i x^i\right) * \left(\sum_{j=0}^{N-1} b_j x^j\right) = \left(\sum_{k=0}^{N-1} c_k x^k\right),$$

onde $c_k = \sum_{i+j=k} a_i b_j$.

Vamos definir um conceito importante para, posteriormente, enxergarmos a relação entre tal anel convolucional de polinômios e reticulados.

Definição 2.2. Um polinômio $f(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{N-1}x^{(n-1}$ com coeficientes em $\mathbb{Z}_q$ é dito **curto** se existe $1 \leq c \ll q$ tal que para cada i, $|a_i| \leq c$.

O problema considerado difícil, baseado na estrutura de anéis convolucionais de polinômios é o seguinte:

Dado $1 < t < N$ *e duas coleções de valores módulo* q, *dadas por*

$$\{\alpha_1, \alpha_2, \ldots, \alpha_t\} \ e \ \{\beta_1, \beta_2, \ldots, \beta_t\},$$

busca-se encontrar um polinômio f *com grau menor do que* N *tal que* f *seja curto e ainda*

$$f(\alpha_i) = \beta_i,$$

para $1 \leq i \leq t$.

Podemos ainda visualizar esse problema como sendo um problema do vetor mais próximo (CVP). Para isso, considere um polinômio p tal que $deg(p) \leq N-1$ e identifique $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{N-1}x^{(n-1}$ com o vetor $(a_0, a_1, a_2, \ldots, a_{N-1}) \in \mathbb{Z}^N$. Seja $\mathscr{L}$ o reticulado de todos os vetores p tais que

$$p(\alpha_i) \equiv 0 (mod\ q),$$

para todo $1 \leq i \leq t$.

Seja F um polinômio, não necessariamente curto, que satisfaz:

$$F(\alpha_i) \equiv \beta_i (mod\ q),$$

para todo $1 \leq i \leq t$. Então,F_0 é o ponto do reticulado $\mathscr{L}$ mais próximo a F, com uma grande possibilidade de $F - F_0$ ser um polinômio curto considerando as avaliações adequadas.

O algoritmo *LLL*, que é utilizado para resolver de forma aproximada o problema do vetor mais curto e também o problema do vetor mais próximo, apresenta bons resultados para a resolução do problema apresentado acima para dimensões menores do que 100. Pesquisadores como Hoffstein (9) estudam até o momento maneiras de mensurar até qual ponto o algoritmo *LLL* se apresenta eficaz na resolução desse problema.

Para a maioria dos polinômios curtos f, existe f^{-1} com a propriedade

$$f * f^{-1} \equiv 1 (mod\ q, x^N - 1).$$

Se os coeficientes de f forem escolhidos entre $\{-1, 0, 1\}$, os coeficientes de f^{-1} serão completamente aleatórios módulo q. Além disso, se g é um outro polinômio curto, os coeficientes de $h = 3f^{-1} * g$ também serão aleatórios. Isso sugere que se m e r forem dois polinômios curtos, m pode ser oculto pela expressão:

$$e = r * h + m,$$

que também é aleatório.

Se multiplicarmos tal expressão por f, temos:

$$a = f * e = f * (r * h + m) = f * (r * (3f^{-1} * g) + m) = f * (3r * f^{-1} * g + m) = 3r * g + f * m,$$

que é um polinômio curto.

O original $3r * g + f * m$ sem considerar sua redução módulo q será recuperado. Reduzindo por $(mod\ 3)$ obtemos $f * m(mod\ 3)$ e multiplicando por $f^{-1}(mod\ 3)$, encontraremos a mensagem m.

O criptossistema NTRU é uma generalização dessa ideia. O problema difícil é: dado um polinômio curto h de grau $N-1$ com coeficientes em $\mathbb{Z}_q$, encontrar um polinômio f com a propriedade de, após a redução módulo q, $f * h$ também seja um polinômio curto.

Esse problema foi rapidamente traduzido para a estrutura de reticulados, resultando em buscar o vetor (f,g) curto em um certo reticulado $2N-$ dimensional. Acredita-se que esse problema seja difícil, mas não se tem ideia do quão difícil ele possa ser.

Vamos conhecer então um pouco mais sobre a estrutura de tal reticulado $2N-$ dimensional e tratar aqui da adaptação do criptossistema NTRU usando reticulados como base ao invés de anéis convolucionais. Para tanto, apresentaremos agora algumas definições e conceitos para tornar a explicação da estrutura do algoritmo NTRU mais clara e didática possível.

A eficiência de um sistema criptográfico baseado em reticulados pode ser melhorada substituindo matrizes gerais por matrizes com uma estrutura especial, de acordo com 14. Tal matriz especial pode ser definida como:

Definição 2.3. Seja $A \in \mathbb{Z}_q^{n\times m}$ uma matriz qualquer. Tal matriz pode ser substituída por uma matriz de blocos:

$$A = (A^{(1)}|\ldots|A^{(m/n)}),$$

onde cada bloco $A^{(i)} \in \mathbb{Z}_q^{n\times n}$ é uma **matriz circulante** na forma:

$$A^{(i)} = \begin{pmatrix} a_1^{(i)} & a_n^{(i)} & \ldots & a_3^{(i)} & a_2^{(i)} \\ a_2^{(i)} & a_1^{(i)} & \ldots & a_4^{(i)} & a_3^{(i)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n-1}^{(i)} & a_{n-2}^{(i)} & \ldots & a_1^{(i)} & a_n^{(i)} \\ a_n^{(i)} & a_{n-2}^{(i)} & \ldots & a_2^{(i)} & a_1^{(i)} \end{pmatrix}.$$

Ou seja, uma matriz circulante é uma matriz cujas colunas são desvios cíclicos da primeira coluna $a^{(i)} = (a_1^{(i)}, \ldots, a_n^{(i)})$.

Podemos ainda representar uma matriz circulante utilizando notação matricial, escrevendo $A^{(i)} = (a^{(i)}, Ta^{(i)}, \ldots, T^{n-1}a^{(i)})$, onde:

$$T = \left(\begin{array}{c|c} 0^T & 1 \\ I & 0 \end{array}\right). \tag{2}$$

Tal estrutura de matriz circulante reduz o armazenamento dos elementos das chaves de mn elementos para apenas m elementos e reduz também o tempo do algoritmo que estará utilizando tal matriz, segundo 14.

Seja a transformação linear T conforme apresentada em 2 que representa a rotação ou desvio cíclico das coordenadas de um vetor e defina $T^*v = (v, Tv, \ldots, T^{n-1}v)$ como sendo a matriz circulante de um vetor $v \in \mathbb{Z}^n$.

Os reticulados usados no criptossistema $NTRU$, são chamados de reticulados convolucionais modulares e podem ser definidos como:

Definição 2.4. Um **reticulado convolucional modular** é um reticulado de dimensão par $2n$ que satisfaz as seguintes propriedades:

i) É fechado com relação a transformação linear que mapeia o vetor (x, y) (com x, y vetores $n-$dimensionais) em (Tx, Ty), que representa o vetor obtido pela rotação cíclica das coordenadas dos vetores x e y.

ii) É um reticulado $q-$ário, no sentido de que sempre contém $q\mathbb{Z}^{2n}$ como subreticulado e ainda, o fato de (x, y) pertencer ao reticulado depende somente de $(x, y) mod\ q$.

Os parâmetros do sistema NTRU são: uma dimensão prima n, um inteiro módulo q, um inteiro pequeno e primo p e uma cota inteira para o peso d_f. Vamos assumir que q é uma potência de 2, ou seja $q = 2^m$, já que estamos apresentando os parâmetros propostos em 12.

Vamos descrever em detalhes como funciona o algoritmo NTRU para depois apresentá-lo em sua estrutura resumida, como feito com os demais algoritmos tratados neste trabalho.

A chave privada usada no criptossistema NTRU é um vetor curto $(f, g) \in \mathbb{Z}^{2n}$. O reticulado associado à chave privada (f, g) e ao parâmetro q é $\mathscr{L}_q((T^* f, T^* g)^T)$, que pode ser visto como o menor reticulado convolucional modular que contém (f, g). Os vetores secretos f, g estão sujeitos às seguintes restrições auxiliares:

- a matriz $(T^* f)$ deve ser invertível módulo q.
- $f \in e_1 + \{p, 0, -p\}^n$ e $g \in \{p, 0, -p\}^n$ são polinômios escolhidos aleatoriamente tais que $f - e_1$ e g possuam exatamente $d_f + 1$ entradas positivas, d_f entradas negativas e as demais $N - 2d_f - 1$ entradas restantes serão nulas.

Tais cotas e restrições são motivadas por razões de eficiência computacional, como o cálculo das chaves públicas, encriptação e decriptação, por exemplo.

A chave pública, devido as propriedades estruturais dos reticulados convolucionais modulares e às restrições para f, é dada por:

$$H = \begin{pmatrix} I & O \\ T^* h & q \cdot I \end{pmatrix},$$

onde $h = (T^* f)^{-1} g\ (mod\ q)$. Para facilitar a notação, podemos apresentar a chave pública informando somente o vetor $h \in \mathbb{Z}_q^n$.

Para encriptar uma mensagem $m \in \{1, 0, -1\}^n$ com exatamente $d_f + 1$ entradas positivas e d_f entradas negativas, procedemos da seguinte maneira: o vetor m é agrupado a um vetor arbitrário $r \in \{1, 0, -1\}^n$ que também possui $d_f + 1$ entradas positivas e d_f entradas negativas, para obter um pequeno vetor de erro $(-r, m) \in \{1, 0, -1\}^{2n}$. Reduzindo o vetor erro $(-r, m)$ módulo a base pública H, obtemos:

$$\begin{pmatrix} -r \\ m \end{pmatrix} mod \begin{pmatrix} I & O \\ T^* h & q \cdot I \end{pmatrix} = \begin{pmatrix} 0 \\ (m + (T^* h) r)\ mod\ q \end{pmatrix}.$$

Como as primeiras n coordenadas desse vetor são sempre nulas, elas podem ser omitidas, deixando somente o vetor $n-$dimensional $c = m + (T^* h) r\ mod\ q$ como texto cifrado.

A decriptação, por sua vez, se dá conforme segue: o texto cifrado c é decriptado multiplicando-o pela matriz secreta $(T^* f)$ módulo q, e obtemos:

$$(T^* f) c\ mod\ q = (T^* f) m + (T^* f)(T^* h) r\ mod\ q = (T^* f) m + (T^* g) r\ mod\ q,$$

onde usamos a identidade $(T^* f)(T^* h) = (T^*(T^* f) h)$ que é válida para quaisquer vetores f e h.

Esse sistema então, se resume ao esquema apresentado pelo Algoritmo 4.2:

Algoritmo 3 O sistema criptográfico de chave pública NTRU

Defina os parâmetros: número primo n, um inteiro módulo q, e uma cota inteira para o peso d_f. (Note que escolhemos no exemplo anterior $p = 3$ para simplificar os cálculos.)
Chave privada: Vetores $f \in e_1 + \{p, 0, -p\}^n$ e $g \in \{p, 0, -p\}^n$, tais que tanto $f - e_1$ quando g contenham exatamente $d_f + 1$ entradas positivas e d_f entradas negativas.
Chave pública: Vetor $h = (T^* f)^{-1} g \ (mod\ q) \in \mathbb{Z}_q^n$.
Encriptação de uma mensagem $m \in \{1, 0, -1\}^n$: Tome um vetor arbitrário $r \in \{1, 0, -1\}^n$, com m e r contendo $d_f + 1$ entradas positivas e d_f entradas negativas. A função de encriptação devolve um vetor c tal que $c = m + (T^* h) r \ mod\ q$.
Decriptação de um vetor $c \in \mathbb{Z}_q^n$: Temos como resultado $(((T^* f) c \ mod\ q) \ mod\ p$, onde a redução módulo q e p produzem vetores com coordenadas em $[q/2, q/2]$ e $[-p/2, p/2]$, respectivamente.

Para parâmetros apropriados, recuperar uma mensagem m de um texto cifrado c é equivalente a encontrar um vetor em um reticulado $\mathscr{L}$ que está mais próximo do vetor $[0, c]$. A dificuldade em resolver esse problema CVP pode ser estimada experimentalmente, segundo 18.

Referências

[1] Ajtai, M. e Dwork, C. *A public-key cryptosystem with worst-case/average-case equivalence*. 1997.

[2] Ajtai, M., *The Shortest Vector Problem in L2 is NP-hard for Randomized Reductions (Extended Abstract)*. In: Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, p. 10-19. ACM. 1998. Disponível em : `http://doi.acm.org/10.1145/276698.276705`

[3] Babai, L., *On Lovász' Lattice Reduction and the Nearest Lattice Point Problem (Shortened Version)*. In: Proceedings of the 2Nd Symposium of Theoretical Aspects of Computer Science, p. 13-20. 1985. Disponível em : `http://dl.acm.org/citation.cfm?id=646502.696106`

[4] Barreto, P.; Biasi, F.P.; Dahab, R.; *et al. Introdução à criptografia pós-quântica*. 2013. Disponível em : `http://dainf.ct.utfpr.edu.br/~maziero/lib/exe/fetch.php/ceseg:2013-sbseg-mc2.pdf`

[5] Boas, P. van Emde , *Another NP-Complete Problem and the Complexity of Computing Short Vectors in a Lattice*. Mathematische Instituut, Universiry of Amsterdam, 1981. Disponível em : `http://turing.wins.uva.nl/~peter/`

[6] Campello, A. *Notas de aula: reticulados, teoremas de Minkowski revisitados e teoremas de somas de quadrados*. 2014. Disponível em : `http://www.ime.unicamp.br/~campello/geometria/reticulados.pdf`

[7] Gamma, N.; Nguyen, P.Q. e Regev, O. *Lattice Enumeration using Extreme Pruning*. In: Advances in Cryptology - Proceedings of EUROCRYPT '10. Springer. 2010.

[8] Goldreich, O.; Goldwasser, S. e Halevi, S. *Public-Key Cryptosystems from Lattice Reduction Problems*. In: Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology, p.112-131. Springer-Verlag, 1997.

[9] Hoffstein, J.. *A history of the development of NTRU, in: EUROCRYPT 2014, Copenhagen*. 2014. Disponível em : `http://http://ec14.compute.dtu.dk/talks/6.pdf`

[10] Hoffstein, J.; Pipher, O. e Silverman, J.H., *An Introduction to Mathematical Cryptography*. Springer Publishing Company, Incorporated, 2008.

[11] Hoffstein, J.; Pipher, O. e Silverman, J.H., *NTRU: A Ring-Based Public Key Cryptosystem*. Springer-Verlag, 1998.

[12] Hoffstein, J.; Pipher, O. e Silverman, J.H., *Hybrid lattice reduction and meet in the middle resistant parameter selection for NTRU-Encrypt*. 2007. Disponível em : `http://grouper.ieee.org/groups/1363/lattPK/submissions.html2007-02`

[13] Hoffstein, J.; Howgrave-Graham, N. e Pipher, J., *et al. NTRUSign: Digital Signatures Using the NTRU Lattice*. Springer-Verlag, 2002.

[14] Micciancio, D. e Regev, O. *Lattice-based Cryptography*. In: Post-Quantum Cryptography - Bernstein, D.J.; Buchmann, J. e Dahmen, E., 2009.

[15] Minkowski, H. *Geometrie der Zahlen*. Leipzig, 1896.

[16] Nguyen, P.Q.; Jacques, S., *Cryptanalysis of the Ajtai-Dwork Cryptosystem*. In: CRYPTO, Lecture Notes in Computer Science, p. 223-242. 1998.

[17] Patarin, J. e Goubin, L. *Trapdoor One-Way Permutations and Multivariate Polynomials*. In: Proc. of ICICS'97, LNCS 1334, p. 356-368. Springer, 1997.

[18] Silverman, J.H., *An introduction to the theory of lattices and applications to cryptography*. 2006. Disponível em : `http://www.math.brown.edu/~jhs/Presentations/WyomingLattices.pdf`