MS877 - Projeto Supervisionado II

Projeto: Reticulados: métodos de decodificação e criptografia

Período 01/08/2010 à 10/12/2010 (*)

Aluno: Franz Pietz

Orientadora: Profª. Dra. Sueli Irene Rodrigues Costa

Instituto de Matemática, Estatística e Computação Científica

Universidade Estadual de Campinas

(*) Trabalho desenvolvido com o suporte do Projeto de Iniciação Científica Bolsa FAPESP - 2008 / 06171-3, com período de 01/08/2008 à 31/12/2010

Sumário

Resumo de Projeto Original

O propósito é o de introduzir o aluno à teoria de códigos corretores de erros e a tópicos de geometria discreta associados. Naturalmente o objetivo subjacente é o de complementar sua formação em matemática nestas subáreas e propiciar sua interação com um grupo de pesquisa ativo. Este projeto está vinculado ao Projeto Temático FAPESP "Teoria da Informação e Códigos (2007/56052-8) com vigência de 04/2008 a 03/2012, coordenado pela orientadora. Integram este projeto 9 pesquisadores de Matemática, Eng. Elétrica e Computação, pesquisadores colaboradores e alunos de pós-graduação da FEEC, do IMECC e do IC da Unicamp.

Introdução - Descrição geral dos temas abordados no projeto

A teoria de códigos corretores de erros tem como objetivo conseguir sistemas de codificação que permitam uma maior confiabilidade nos canais de transmissão de informação. Ela integra a grande área do conhecimento chamada de "Teoria da Informação", cujo marco inicial é o famoso artigo de C.E. Shannon intitulado "A Mathematical Theory of Communications" (1948) que influenciou o desenvolvimento de teorias matemáticas relacionadas à tecnologias de comunicação e informação. Desde a publicação do artigo, diversas sub-áreas da matemática têm sido utilizadas para a resolução de problemas relacionados à Teoria da Informação (dentre elas a Álgebra Linear de corpos finitos e Geometria Discreta), onde destacamos os códigos lineares e os reticulados, objetos de estudo deste prejeto de iniciação científica. Códigos lineares são, grosso modo, códigos obtidos pela adição de redundância (com propriedades lineares) em um conjunto de "palavras" de um certo tamanho em um alfabeto. De maneira mais precisa, são códigos gerados por transformações lineares injetivas entre dois espaços vetoriais sobre corpos finitos. São fortemente utilizados na prática por herdarem todas as propriedades de espaços vetoriais, o que facilita a busca por códigos "bons" (que corrigem/detectam o maior número de erros com a menor redundância) e o desenvolvimento de algoritmos de codificação/decodificação. Dentre os códigos lineares, destacam-se os códigos de Hamming, Golay e Reed-Solomon, que podem ser vistos em [2].

1 Tópicos desenvolvidos no semestre

Inicialmente foram estudados o capítulo 2 de [1] e os capítulos 2 e 4 de [8], que apresentam os fundamentos básicos e os tipos mais conhecidos de reticulados. Para uma introdução em técnicas de decodificação em reticulados, foram estudados tópicos dos artigos [10], [11] e [12], que tratam do problema da decodificação esférica, o problema de mínimos quadrados com variáveis inteiras e apresentam uma proposta de algoritmo para obter soluções para a decodificação esférica.

Para a seção de criptografia foi estudado o capítulo 5, que apresenta uma proposta para criptografia baseada em reticulados.

2 Metodologia

A metodologia utilizada foi o estudo individual dos capítulos dos livros e artigos citados, resolução de questões para fixação das idéias e aprofundamento de tópicos, além de reuniões regulares com o professora orientadora para discussão dos

temas. Foi utilizado, também, o espaço e recursos do Laboratório Matemática Discreta e Códigos, IMECC - Unicamp, incluindo contato com alunos de pós-graduação que desenvolvem pesquisas em áreas relacionadas ao projeto.

Participei dos seminários com a presença da orientadora, do Prof. Dr. Cristiano Torezzan e dois alunos de Iniciação Científica do IC, Thomaz Milani e Rafael Stafocher, nos quais apresentei a proposta do algoritmo e discuti sobre a decodificação esféricas. Além disso, tive contato frequente com os alunos de doutorado envolvidos no projeto temático.

3 Reticulados e decodificação via esferas

Nessa seção resumimos os conceitos básicos de reticulados, usando como referências [1] e [8], e abordamos o métodos de decodificação via esferas, estudado tópicos de [9], [10], [11] e [12].

3.1 Conceitos básicos

O problema de encontrar o melhor código em $\mathbb{Z}_2^n$ está associado ao problema de **empacotamento esférico**, que trata da distribuição de esferas de raio r de maneira que duas esferas se toquem em apenas um ponto da casca ou não haja intersecção, ocupando a maior área possível[1]. O centro dessas esferas pode ser tomado em pontos de um reticulado. Um subconjunto Λ de $\mathbb{R}^n$ é um **reticulado** se existe uma base $\beta = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n\}$ de $\mathbb{R}^n$ tal que $\mathbf{x} \in \Lambda$ se, e somente se, $\mathbf{x} = a_1\mathbf{u}_1 + a_2\mathbf{u}_2 + \ldots + a_n\mathbf{u}_n$ com $a_i \in \mathbb{Z}$ para todo i. A base de um reticulado não é única e a matriz mudança de base M entre duas bases tem coeficientes inteiros e determinante ± 1.

As figuras a seguir mostram dois exemplos de reticulados:

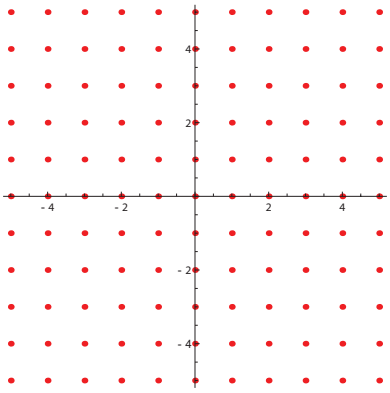

Figura 1: $\beta = \{(0,1),(1,0)\}$.

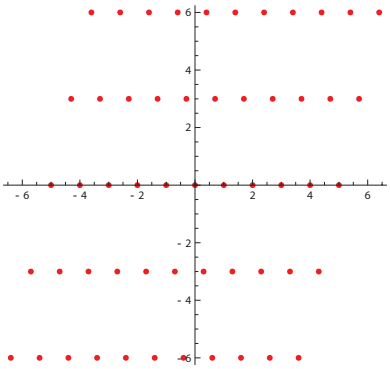

Figura 2: $\beta = \{(0,1),(0.7,3)\}$.

Definimos como **raio de empacotamento** o maior raio possível para as esferas centradas nos pontos dos reticulados sem que haja sobreposição das mesmas. Para a base $\beta = \{(0,1),(1,0)\}$, por exemplo, temos o reticulado $\mathbb{Z}^2$ e a distância entre os pontos do reticulado é exatamente 1, ou seja, o maior raio de empacotamento é $\rho = \dfrac{1}{2}$.

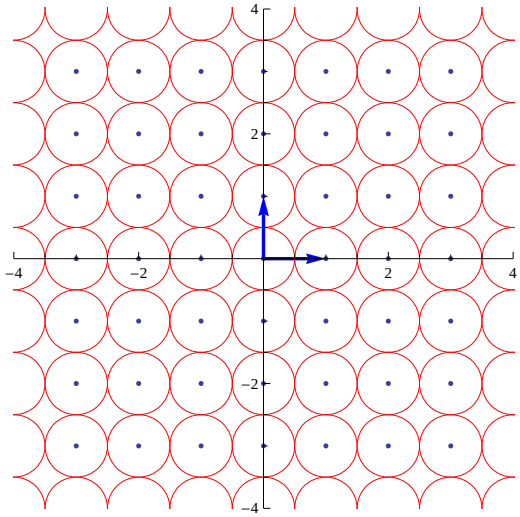

Figura 3: Empacotamento no reticulado $\mathbb{Z}^2$.

O politopo fundamental de um reticulado é o sólido definido por $P = \left\{ \sum_{i=0}^{n} a_i \mathbf{u}_i,\ 0 \leq a_i \leq 1 \right\}$, ou seja, é um sólido formado pelas combinações lineares dos vetores da base. Um exemplo desse sólido é mostrado na figura a seguir.

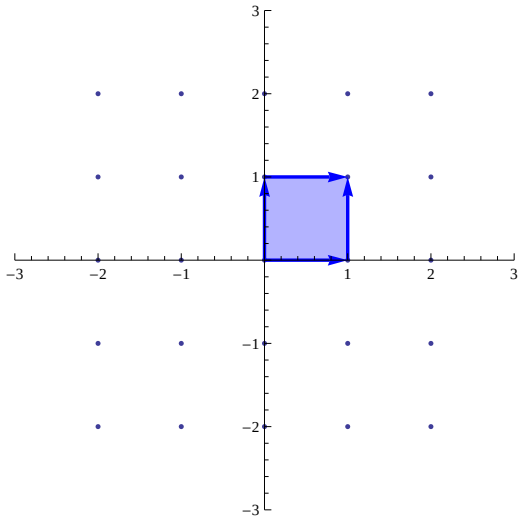

Figura 3: Politopo fundamental.

Podemos avaliar a densidade do reticulado, ou seja, o quanto o plano está sendo coberto pelo empacotamento esférico, calculando a razão entre o raio da esferas e a área do politopo fundamental. O reticulado mais denso para o $\mathbb{R}^2$ é o reticulado A_2, também chamado reticulado hexagonal, que possui base $\beta = \{(1,0), (1/2, \sqrt{3}/2)\}$ e densidade $0,9069$.

Definimos a matriz geradora de um reticulado como uma matriz A cujas colunas são os vetores da base que gera o mesmo. Chamamos a matriz $G = A^T A$ de matriz de Gram do reticulado. Como um reticulado pode ser gerado por mais de uma base, podemos ter diferentes matrizes de Gram para um mesmo reticulado. Entretanto, o determinante dessas matrizes é sempre o mesmo e depende apenas do reticulado. Um resultado importante é que $\det(A) = vol(P)^2$. Assim, podemos calcular a densidade de um reticulado por

$$\Delta = \frac{vol(B_p(0))}{\sqrt{\det(A)}}$$

onde Δ é a densidade e $B_p(0)$ é a bola (esfera em dimensões superiores) de centro zero.

Decodificar em reticulados é, basicamente, encontrar o ponto do reticulado mais próximo à um ponto y recebido. Para reticulados gerais, esse é um problema NP-Difícil. Existem algumas técnicas para diminuir a complexidade desse problema, sendo uma delas a decodificação via esferas, que será detalhada na seção a seguir.

3.2 Decodificação via esferas em reticulados

Nessa seção apresentaremos o método de decodificação chamado **decodificação via esferas** (Sphere Decoding, no original), apresentado por Fincke e Pohst [9] em 1985. Consiste, basicamente, em encontrar todos os pontos localizados dentro de uma esfera S de raio d e centro em y, e encontrar o ponto em S mais próximo de y. A busca dentro de uma esfera reduz o número de operações necessárias para decodificar um ponto. O principal passo desse método é a escolha do raio da esfera [10, 11] e existem alguns problemas relacionados, como:

- Escolha do raio d: um raio muito pequeno pode gerar uma esfera que não contém pontos do reticulado; um raio muito grande pode gerar uma esfera com muitos pontos, aumentando a complexidade do cálculo.
- Listar os pontos que estão dentro de uma esfera: requer o cálculo da distância de cada ponto do reticulado ao centro da esfera. Uma escolha natural para o raio d seria o raio de empacotamento do reticulado [10], definido anteriormente. Entretanto, o calculo do raio de empacotamente de um reticulado genérico é considerado um problema NP-dificil.

Um método de resolução do problema é tratá-lo como um problema de mínimos quadrados inteiros.

3.2.1 Mínimos quadrados inteiros

O problema de **mínimos quadrados inteiros** aparece em aplicações de diversas áreas, tais como comunicações, criptografia e GPS. É definido por:

$$\min_{s\in\mathbb{R}^m} \|Hs - y\| \qquad (1),$$

com $y \in R^n$ e $H \in R^{n\times m}$, sendo s inteiro e y e H reais.

Utilizando a fatoração QR em H, geramos $H = Q\begin{bmatrix} R \\ 0 \end{bmatrix}$, com Q em $\mathbb{R}^{n\times n}$ ortogonal e R em $\mathbb{R}^{m\times n}$ triangular superior. Aplicando QR em (1), temos [11]:

$$\begin{aligned}
&\|Hs - y\|_2^2 \\
&= \left\| Q\begin{bmatrix} R \\ 0 \end{bmatrix} s - y \right\|_2^2 \\
&= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} s - Q^T y \right\|_2^2 \\
&= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} s - \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} y \right\|_2^2 \\
&= \left\| Rs - Q_1^T y \right\|_2^2 + \left\| Q_2^T y \right\|_2^2
\end{aligned}$$

Podemos ver que o segundo termo independe de s, reduzindo (1) para

$$\min_{s\in\mathbb{R}^m} \|Rs-\hat{y}\|_2^2 \qquad (2),$$

onde $\hat{y} = Q_1^T y$.

Para o problema de decodificação esférica, queremos calcular $\min_{s\in\mathbb{R}^m} \|Rs-\hat{y}\|_2^2$ com $s \in S\{s \mid \|Rs-\hat{y}\|_2 \leq d\}$, mas para isso, precisamos encontrar um raio d que permita resolver o problema.

3.2.2 Estimativa do raio da esfera (resumo)

Como mostrado no relatório anterior, em [12] é apresentado um método determinístico de estimar um raio d da esfera S, utilizando a estimativa de Babai. Como o cálculo desse raio pode sofrer desvios devido ao arredondamento da máquina, levando a necessidade de adicionar uma folga durante os cálculos. A ideia principal da estimativa é descrita a seguir.

- **Entradas**: Matriz R triangular superior e $\hat{y} = Q_1^T y$, obtido a partir de y reduzido pela fatoração QR.
- **Saida:** O raio de busca $\hat{d}$

1. resolva $s \in \mathbb{R}^m$ em $Rs = \hat{y}$
2. arredonde $\hat{s} = \lceil s \rfloor$
3. defina $\hat{d} = \|R\hat{s} - y\|_2$

3.2.3 Descriçao do algoritmo

Após obter o raio inicial, sabemos que $\hat{d}^2 \geq \|Rs-\hat{y}\|_2^2$. Como R é triangular superior, podemos escrever a inequação como $\hat{d}^2 \geq \sum_{i=1}^{m} \left(\sum_{j=1}^{m} r_{i,j}s_j - \hat{y}_i \right)^2$, onde $r_{i,j}$representa a posição (i,j) da matriz R.

Expandindo o somatório, temos $\hat{d}^2 \geq (\hat{y}_m - r_{m,m}s_m)^2 + (\hat{y}_{m-1} - r_{m-1,m}s_m - r_{m-1,m-1}s_{m-1})^2 + \ldots$

Podemos ver que o primeiro termo do somatório depende apenas da m-ésima coordenada do ponto s, o segundo termo depende da m e $(m-1)$-ésima entradas, e assim por diante. Assim, para que s pertenca à esfera S, é necessário que $\hat{d}^2 \geq (\hat{y}_m - r_{m,m}s_m)^2$. Assim, temos que $\left\lfloor \frac{-\hat{d}+\hat{y}_m}{r_{m,m}} \right\rfloor \leq s_m \leq \left\lceil \frac{\hat{d}+\hat{y}_m}{r_{m,m}} \right\rceil$, ou seja, temos um intervalo de valores válidos para s_m.

Para cada valor de s_m , definimos $\hat{d}^2_{m-1} = \hat{d}^2_{m-1} - (\hat{y}_m - r_{m,m}s_m)^2$ e $\hat{y}'_{m-1} = \hat{y}_{m-1} - r_{m-1,m}s_m$. Temos que, se $\hat{d}^2_{m-1} \geq \left(\hat{y}'_{m-1} - r_{m-1,m-1}s_{m-1}\right)^2$, então podemos definir também um intervalo para $s_m - 1$, dado por

$$\left\lfloor \frac{\hat{d} + \hat{y}_{m-1}}{r_{m-1,m-1}} \right\rfloor \leq s_{m-1} \leq \left\lceil \frac{-\hat{d}_{m-1} + \hat{y}_{m-1}}{r_{m-1,m-1}} \right\rceil$$

Repetindo esse procedimento, podemos obter os intervalos para todas as coordenadas de s. Após obter o valor de s_1, podemos calcular a norma de cada vetor s e obter, assim, o ponto do reticulado mais próximo do ponto y recebido e, finalmente, realizar a decodificação.

A desvantagem desse método está na necessidade de calcular um intervalo para cada valor do intervalo obtido para o nível anterior. Ao final do processo, temos uma árvore, a qual chamaremos de árvore de estados, onde cada folha (nó sem filhos) representa um valor de s_1 . Podemos calcular a norma de cada vetor obtido percorrendo a árvore de cada folha até a raiz.

Encontrar todos os valores de s_1 é custoso. Sabemos que a árvore de estados gerada ao final do algoritmo possui altura n, mas não podemos dizer ao certo a quantidade de nós gerados para os calculos. Chamando de k o tamanho do maior intervalo gerado durante os calculos e considerando, como pior caso, que todos os intervalos obtidos tenham tamanho k, teremos que calcular k^n possiveis valores de s_1, além de ter que calcular a menor norma para todas essas possibilidades. O algoritmo para encontrar todos os valores possíveis é claramente exponencial, tornando-se inviável para dimensões grandes.

Para melhorar a performance do algoritmo proposto, podemos utilizar uma estratégia de *backtracking*, onde realizamos uma busca em profundidade na árvore de estados, seguindo o seguinte critério:

- Inicialmente, utilizamos o raio de busca definido como anteriormente.
- Escolhemos o menor valor do intervalo calculado no nível atual para gerar o intervalo para o nivel seguinte, até chegarmos no nível de s_1. Dessa maneira, teremos calculado todas as coordenadas de um ponto do reticulado e poderemos calcular a distância dele para o ponto recebido, armazenando esse valor. Com esse procedimento, chegamos ao extremo mais a esquerda da árvore de estados, que representa o primeiro valor possivel para um intervalo.
- Começaremos a calcular os valores encontrados para os intervalos restantes, fazendo como limitante superior a distância calculada anteriormente, ou seja, podemos calcular uma distância parcial entre o ponto recebido e as coordenadas já calculadas para o nível atual, sendo que amadureceremos um nó (calcularemos seus filhos) somente se a distância parcial for menor à distância obtida na primeira etapa.
- Se ao chegar no nível de s_1 obtivermos um valor menor que a distância já calculada, atualizamos o valor do limitante superior e continuamos calculando os intervalos, seguindo o passo anterior.

Ao final do processo, teremos que o limitante superior é a menor distância calculada, que corresponde ao ponto mais próximo do ponto recebido, ou seja, finalizamos a decodificação.

A vantagem de métodos como o backtracking é que podemos eliminar o cálculo desnecessário de vetores de norma grande e que seriam descartados no final do processo, economizando assim recursos computacionais e diminuindo o tempo de execução do algoritmo, tornando a decodificação mais eficiente.

3.2.4 Resultados obtidos

Nessa seção, apresentaremos alguns resultados obtidos para reticulados em $\mathbb{R}^2$ e $\mathbb{R}^3$, para melhor compreensão do algoritmo.

Podemos estimar o número de pontos dentro da esfera de busca calculando a razão entre a área (volume) da circunferência (esfera, bola) sobre a raiz do determinante da matriz de Gram do reticulado (volume do politopo fundamental). As figuras com os resultados seguem o seguinte padrão:

- o ponto recebido y é representado em vermelho;
- os pontos que o algoritmo verifica são indicados em verde;
- os ponto mais próximo é representado em cinza;
- os vetores em laranja representam a caixa de busca dos pontos;
- os vetores em azul representam a base do reticulado;

Reticulado 1

- Base: $\beta = \{(1, 0), (1/2, \sqrt{3}/2)\}$
- Ponto recebido: (0,3,0,8)
- Raio inicial: 0,210617
- Número de pontos dentro do disco: 1
- Pontos verificados: (-1,1), (0,1) (na base do reticulado)
- Ponto mais próximo: (0,1)

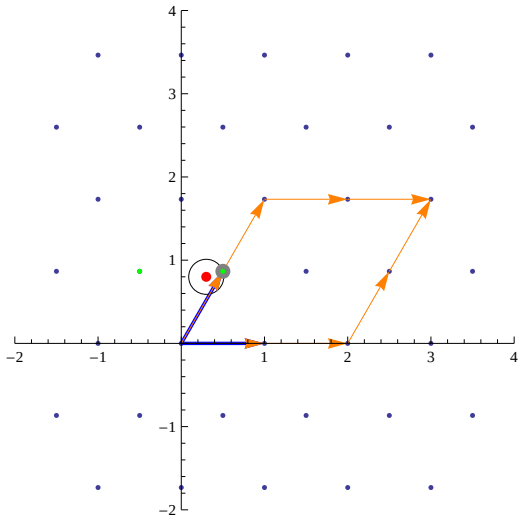

Figura 4: Decodificação para o reticulado 1.

- Ponto recebido: (1,1)
- Raio inicial: 0,517638
- Número de pontos dentro do disco: 2
- Pontos verificados: (0,1), (1,1) (na base do reticulado)
- Ponto mais próximo: (0,1)

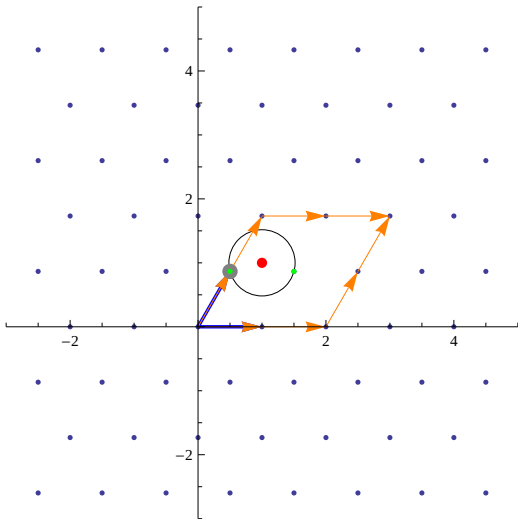

Figura 5: Decodificação para o reticulado 1.

Os pontos verificados estão equidistantes do ponto recebido. Dessa maneira, deveriamos solicitar que a mensagem seja retransmitida, para criar um critério de desempate e decodificar de maneira correta. No caso do algoritmo desenvolvido, ele escolhe o primeiro ponto na lista de pontos encontrados.

Reticulado 2

- Base: $\beta = \{(2,1),(5,3)\}$
- Ponto recebido: (1,3 , 1,8)
- Raio inicial: 1,52643
- Número de pontos dentro do disco: 8
- Pontos verificados:(3, -1), (4, -1), (1, 0), (-2, 1), (-7, 3), (-12, 5) (na base do reticulado)
- Ponto mais próximo: (-7,3)

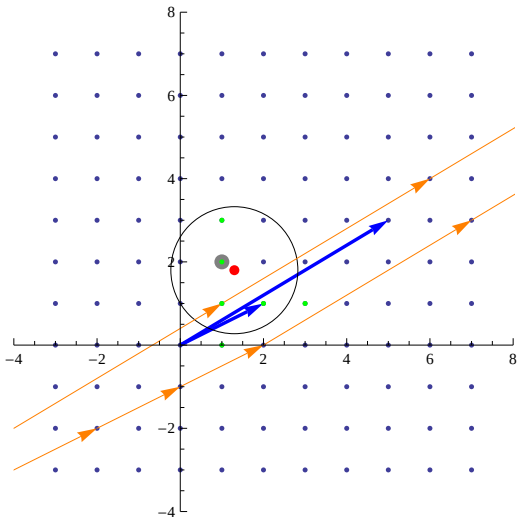

Figura 4: Decodificação para o reticulado 1.

Esse exemplo mostra que, conforme o ângulo entre os vetores do reticulado diminui, mais pontos são verificados. Esse fato serve como base para propostas de criptosistemas baseados em reticulados, como será apresentado na seção 3.3.

Reticulado 3

- Base: $\beta = \{(1, 0.5, 0), (0, \sqrt{3}/2, 0), (0, 0, 1)\}$
- Ponto recebido: (1,3,2.1)
- Raio inicial: 0.649263
- Número de pontos dentro do disco: 6
- Pontos verificados:(-1, 3, 2), (0, 3, 2), (1, 3, 2), (-2, 4, 2), (-1, 4, 2), (0, 4, 2) (na base do reticulado)
- Ponto mais próximo:(-1, 4, 2)

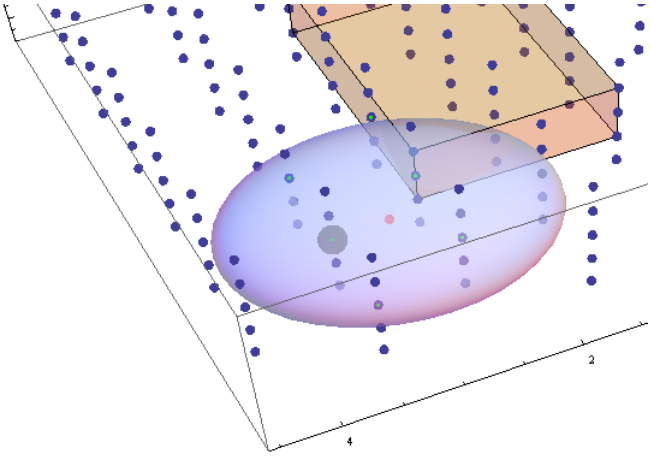

Figura 5: Decodificação para o reticulado 3.

Podemos notar que o número de pontos cresce consideravelmente já para a dimensão 3. Podemos adiantar que para dimensões grandes, o número de pontos para verificar é muito grande. Dessa maneira, o uso da estratégia de backtracking é realmente aconselhável.

Reticulado 4

- Base: $\beta = \{(5, 0, 9), (2, 1, 0), (1, 0, 2)\}$
- Ponto recebido: (1,3,2.1)
- Raio inicial:1.62126
- Número de pontos dentro do disco: 26

- Pontos verificados:(3, -1, -12), (4, -1, -12), (2, -1, -11), (3, -1, -11), (2, -1, -10), (3, -1, -10), (2, -1, -9), (3, -1, -9), (1, 0, -7), (2, 0, -7), (1, 0, -6), (2, 0, -6), (1, 0, -5), (2, 0, -5), (1, 0, -4), (2, 0, -4), (1, 0, -3), (0, 0, -2), (1, 0, -2), (0, 0, -1), (1, 0, -1), (0, 0, 0), (1, 0, 0), (0, 0, 1), (1, 0, 1), (0, 0, 2), (-1, 0, 3), (0, 0, 3), (-1, 0, 4), (0, 0, 4), (-1, 0, 5), (0, 0, 5), (-1, 0, 6), (-2, 0, 7), (-1, 0, 7), (-2, 0, 8), (-1, 0, 8), (-2, 0, 9), (-1, 0, 9), (-5, 1, 24), (-6, 1, 25), (-5, 1, 25), (-6, 1, 26), (-5, 1, 26), (-6, 1, 27), (-5, 1, 27), (-6, 1, 28), (-5, 1, 28), (-6, 1, 29), (-7, 1, 30), (-6, 1, 30), (-7, 1, 31), (-6, 1, 31), (-7, 1, 32), (-6, 1, 32), (-7, 1, 33), (-7, 1, 34), (-9, 2, 42), (-10, 2, 43), (-9, 2, 43), (-10, 2, 44), (-9, 2, 44), (-10, 2, 45), (-9, 2, 45), (-10, 2, 46), (-9, 2, 46), (-10, 2, 47), (-11, 2, 48), (-10, 2, 48), (-11, 2, 49), (-10, 2, 49), (-11, 2, 50), (-10, 2, 50), (-11, 2, 51), (-10, 2, 51), (-12, 2, 52), (-11, 2, 52) (na base do reticulado)

- Ponto mais próximo:(0,0, 2)

Para esse exemplo, não foi utilizada a estratégia de backtracking, para demonstrar a quantidade de pontos verificados por um algoritmo iterativo comum. Nesse caso, foram verificados 77 pontos. Não foi possível gerar uma figura visualizável para esse exemplo, pois a base do reticulado é 'ruim' e seriam necessários muitos pontos para uma boa visualização.

3.3 Utilização da decodificação via esferas em códigos corretores e criptografia

A decodificação via esferas se mostrou eficiente para dimensões pequenas ($<$100). Quando tratamos de códigos corretores de erros, no geral, utiliza-se blocos de informação de tamanho pequeno. Então, transformando o código original em um reticulado, temos um método de decodificação eficiente para códigos corretores de erros de dimensões pequenas.

No Capítulo 5 de [6] , há uma proposta de um sistema criptográfico baseado em reticulados, entitulado GGH, proposto inicialmente em [13], no qual a segurança se baseia na dificuldade de decodificar em um reticulado genérico. Basicamente, esse sistema utiliza como chave privada uma de base "boa" B para um reticulado Λ, formada por vetores curtos e quase ortogonais. Essa escolha permite decodificar de maneira eficiente, pois apresenta uma melhora significativa no cálculo de problemas como encontrar o vetor mais curto, problema computacionalmente dificil para reticulados genéricos. Como chave privada, temos uma base "ruim" (o ângulo entre os vetores é pequeno) H para o mesmo reticulado Λ, ou seja, temos $\Lambda(B) = \Lambda(H)$, de maneira que seja dificil decodificar um ponto recebido utilizando H.

Como visto anteriormente, a decodificação esférica é um método eficiente para bases onde os vetores são quase ortogonais, ou seja, para as bases "boas". Assim, teríamos um método eficiente para encontrar o ponto mais próximo a uma mensagem cifrada pelo GGH. Já para bases "ruim", o algoritmo se mostrou menos eficiente. Como os sistemas criptograficos trabalham com dimensões grandes ($>$ 500), não há garantias que a decodificação via esferas seja uma ameaça contra sistemas baseados em reticulados.

Referências

[1] Lavor C. C., Alves M. M. S, Siqueira R. M e Costa S. I. R, "Uma Introdução̧oo à Teoria de Códigos", Notas em Matemática Aplicada SBMAC vol. 21 2006

[2] A. Hefez e M. A. T. Villela, "Códigos corretores de erros", IMPA, 2002.

[3] F.J. Mac-Williams e N.J.A. Sloane, "Theory of Error Correcting Codes", North-Holland, 1977.

[4] W.C. Huffman e V. Pless, "Fundamentals of Error-Correcting Codes",Cambridge University Press, 2003.

[5] J.I. Hall, "Notes on Coding Tremovendoheory", http://www.mth.msu.edu/%7Ejhall/classes/codenotes/coding-notes.html.

[6] J.I. Hall, "Notes on Coding Tremovendoheory", http://www.mth.msu.edu/%7Ejhall/classes/codenotes/coding-notes.html.Bernstein, D.J.; Buchmann, J; Dahem, E. "Post-Quantum Cryptography", Springer-Verlag, 2009.

[7] Menezes,A.J.; Vanstone,S.A.; van Oorschot, A.C. "Handbook of Applied Cryptography", CRC Press, 2001.

[8] J.H. Conway and N.J.A. Sloane. "Sphere Packings, Lattices and Groups". Springer-Verlag, New York, 1988.

[9] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis", Mathematics of Computation, vol. 44, no. 170, pp. 463-471, Apr. 1985.

[10] B. Hassibi and H. Vikalo, "On the Sphere Decoding Algorithm I. Expected Complexity", IEEE Transactions on Signal Processing, vol. 53, no. 8, pp. 2806-2818, Aug. 2005.

[11] Zhao, F. and Qiao, S. 2009. Radius selection algorithms for sphere decoding. In Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering (Montreal, Quebec, Canada, May 19 - 21, 2009). B. C. Desai and C. K. Leung, Eds. C3S2E '09. ACM, New York, NY, 169-174

[12] Sanzheng Qiao, "Integer least squares: Sphere decoding and the LLL algorithm", Proceedings of C3S2E-08, ACM International Conference Proceedings Series, pp. 23-28, May 2008.

[13] Goldreich, O., Goldwasser, S., and Halevi, S.: Public-key cryptosystems from lattice reduction problems. In Advances in cryptology, volume 1294 of Lecture Notes in Comput. Sci., pages 112–131. Springer (1997).

[14] Micciancio, D.: Improving lattice based cryptosystems using the hermite normal form. In J. Silverman, editor, Cryptography and Lattices Conference — CaLC 2001, volume 2146 of Lecture Notes in Computer Science, pages 126–145. Springer-Verlag, Providence, Rhode Island (2001).