Application of the LLL Algorithm in Sphere Decoding

Sanzheng Qiao

Department of Computing and Software
McMaster University

August 20, 2008
Outline

1. Introduction
   - Application
   - Integer Least Squares

2. Sphere Decoding
   - Reducing Dimension
   - Searching Lattice Points
   - Choosing a Radius

3. The LLL Algorithm

4. Conclusion
A communication channel

\[ x : \text{code vector, integer} \]
\[ A : \text{channel matrix, real} \]
\[ y : \text{received signal, } y = Ax + v \]

\[ \min_{x \in \mathbb{Z}^m} \| Ax - y \|_2^2 \]
Integer Least Squares

\[
\min_{x \in \mathbb{Z}^m} \|Ax - y\|_2^2
\]

- **A**: Generating matrix, \(n\)-by-\(m\), \(n \geq m\), real
- **y**: \(n\)-vector, real
- **x**: \(m\)-vector, solution, integer
A Naive Approach

A seemingly simple approach, Babai solution

\[ x = \lceil A^\dagger y \rceil \]

Example

\[
A = \begin{bmatrix}
1 & 4 \\
2 & 5 \\
3 & 6 \\
\end{bmatrix}
\]
\[
y = \begin{bmatrix}
1 \\
1 \\
1 \\
\end{bmatrix}
\]
A Naive Approach

A seemingly simple approach, Babai solution

$$x = \lceil A^\dagger y \rceil$$

Example

$$A = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix} \quad y = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

real LS solution

$$\begin{bmatrix} -0.3333 \\ 0.3333 \end{bmatrix}$$

rounded to

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

giving residual $\|Ax - y\|_2 = \sqrt{3}.$
The integer least squares solution

\[ x = \begin{bmatrix} -2 \\ 1 \end{bmatrix}, \]

giving residual \( \|Ax - y\|_2 = \sqrt{2} \).
A graph of the naive approach
In general, integer least squares problem is non-polynomial (NP) hard.
Outline

1. Introduction
   - Application
   - Integer Least Squares

2. Sphere Decoding
   - Reducing Dimension
   - Searching Lattice Points
   - Choosing a Radius

3. The LLL Algorithm

4. Conclusion
Problem Setting

1. Search for all lattice points inside the sphere

\[ \|Ax - y\|_2 \leq \rho \]

of radius \( \rho \).

2. Among the lattice points inside the sphere, find the one that minimizes \( \|Ax - y\|_2 \).
Problem Setting

1. Search for all lattice points inside the sphere

\[ \| Ax - y \|_2 \leq \rho \]

of radius \( \rho \).

2. Among the lattice points inside the sphere, find the one that minimizes \( \| Ax - y \|_2 \).

Choosing a radius \( \rho \)

- Too large, too many lattice points inside sphere, expensive
- Too small, no lattice points inside sphere
Reducing Dimension

QR decomposition

\[ A = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R & \ 0 \end{bmatrix} \]

\[ [Q_1 \quad Q_2]: \text{orthogonal} \]

\[ R: \text{upper triangular, } m\text{-by-}m \]
Reducing Dimension

QR decomposition

\[ A = [Q_1 \quad Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix} \]

\([Q_1 \quad Q_2]\): orthogonal

\(R\): upper triangular, \(m\)-by-\(m\)

Then

\[ \|Ax - y\|_2^2 = \|Rx - Q_1^T y\|_2^2 + \|Q_2^T y\|_2^2 \]
Reducing Dimension (cont.)

\[ \| Ax - y \|_2^2 \leq \rho^2 \]

becomes the triangular ILS problem:

\[ \| Rx - \hat{y} \|_2^2 \leq \hat{\rho}^2 \]

\[ \hat{y} = Q_1^T y \]

\[ \hat{\rho}^2 = \rho^2 - \| Q_2^T y \|_2^2 \]
Searching

Partition

\[ R_x - \hat{y} = \begin{bmatrix} R_{1:m-1,1:m-1} & r_{1:m-1,m} \\ 0 & r_{m,m} \end{bmatrix} \begin{bmatrix} x_{1:m-1} \\ x_m \end{bmatrix} - \begin{bmatrix} \hat{y}_{1:m-1} \\ \hat{y}_m \end{bmatrix} \]
Partition

\[
Rx - \hat{y} = \begin{bmatrix}
R_{1:m-1,1:m-1} & r_{1:m-1,m} \\
0 & r_{m,m}
\end{bmatrix} \begin{bmatrix}
x_{1:m-1} \\
x_m
\end{bmatrix} - \begin{bmatrix}
\hat{y}_{1:m-1} \\
\hat{y}_m
\end{bmatrix}
\]

\[
\|Rx - \hat{y}\|_2^2 = \|R_{1:m-1,1:m-1}x_{1:m-1} - (\hat{y}_{1:m-1} - x_mr_{1:m-1,m})\|_2^2 + (r_{m,m}x_m - \hat{y}_m)^2
\]

\[\leq \hat{\rho}^2\]
Searching Lattice Points

Searching (cont.)

Two necessary conditions:

1. \[ |r_{m,m}x_m - \hat{y}_m| \leq \hat{\rho} \]

2. \[ \| R_{1:m-1,1:m-1} x_{1:m-1} - (\hat{y}_{1:m-1} - x_m r_{1:m-1,m}) \|^2 \leq \tilde{\rho}^2, \]
\[
\tilde{\rho}^2 = \hat{\rho}^2 - (r_{m,m}x_m - \hat{y}_m)^2
\]
Two necessary conditions:

1. \(|r_{m,m} x_m - \hat{y}_m| \leq \hat{\rho}\)

2. \(\|R_{1:m-1,1:m-1} x_{1:m-1} - (\hat{y}_{1:m-1} - x_m r_{1:m-1,m})\|_2^2 \leq \tilde{\rho}^2,\)

\(\tilde{\rho}^2 = \hat{\rho}^2 - (r_{m,m} x_m - \hat{y}_m)^2\)

Sphere decoding:
Find all integers satisfying cond1;
For each integer solve cond2 recursively. (DFS)
Choosing $\rho$

Hassibi and Vikalo, 2005

In communications

$$y = Ax + v$$

$v$: white noise, variance $\sigma^2$

Given a probability $p$,

1. Find $\alpha$ satisfying

$$p = \int_0^{\alpha n/2} \frac{\lambda^{n/2-1}}{\Gamma(n/2)} e^{-\lambda} d\lambda$$

2. $\rho^2 = \alpha n \sigma^2$
Choosing $\rho$ (cont.)

- The solution lies in the sphere of radius $\rho$ with probability $\rho$.
- The expected complexity is polynomial, often roughly cubic.
- Works well when $\sigma^2$ is small.
Choosing $\rho$ (cont.)

- The solution lies in the sphere of radius $\rho$ with probability $p$.
- The expected complexity is polynomial, often roughly cubic.
- Works well when $\sigma^2$ is small.
- Channel matrix $A$ is not taken into consideration (assuming some statistical characteristics).
Choosing $\rho$ (cont.)

We propose:

1. Solve for real LS solution $\hat{x} = R^{-1}\hat{y}$

2. $\hat{\rho}^2 = \|R\lfloor\hat{x}\rfloor - \hat{y}\|^2_2$
Choosing $\rho$ (cont.)

We propose:

1. Solve for real LS solution $\hat{x} = R^{-1} \hat{y}$

2. $\rho^2 = \| R\lfloor \hat{x} \rfloor - \hat{y} \|_2^2$

At least one lattice point in sphere, deterministic. Both $R(A)$ and $\hat{y}(v)$ are taken into account.
Choosing $\rho$ (cont.)

We propose:

1. Solve for real LS solution $\hat{x} = R^{-1} \hat{y}$

2. $\hat{\rho}^2 = \| R \lceil \hat{x} \rceil - \hat{y} \|_2^2$

At least one lattice point in sphere, deterministic. Both $R(A)$ and $\hat{y}(v)$ are taken into account.

Error in the computed $R^{-1} \hat{y}$ must be addresses.
1. Introduction
   - Application
   - Integer Least Squares

2. Sphere Decoding
   - Reducing Dimension
   - Searching Lattice Points
   - Choosing a Radius

3. The LLL Algorithm

4. Conclusion
What is the LLL algorithm?

What is the LLL algorithm?


QRZ decomposition

\[ A = QRZ^{-1} \]

- \( Q \): orthonormal columns
- \( Z \): unimodular, integer, \( \det(Z) = \pm 1 \)
- \( R \): upper triangular, reduced
What is the LLL algorithm?


QRZ decomposition

\[ A = QRZ^{-1} \]

- **Q**: orthonormal columns
- **Z**: unimodular, integer, \( \det(Z) = \pm 1 \)
- **R**: upper triangular, reduced

1. \[ |r_{i,j}| \leq |r_{i,i}|/2, \quad j > i \]

2. \[ r^2_{i+1,i+1} \geq \omega r^2_{i,i} - r^2_{i,i+1}, \quad 0.25 \leq \omega \leq 1 \]
What is the LLL algorithm? (cont.)

Application:
Cryptography (integer arithmetic)
What is the LLL algorithm? (cont.)

Application:
Cryptography (integer arithmetic)

Luk and Tracy (2008), floating-point
Integer Gram-Schmidt scheme?
Combination of Givens reflection and integer Gaussian reduction.
What is the LLL algorithm? (cont.)

Application:
Cryptography (integer arithmetic)

Luk and Tracy (2008), floating-point
Integer Gram-Schmidt scheme?
Combination of Givens reflection and integer Gaussian reduction.

Luk and SQ (2007), numerical properties
What does the LLL algorithm do?

Example ($\omega = 0.75$)

$$
\begin{bmatrix}
1 & 4 \\
2 & 5 \\
3 & 6
\end{bmatrix} = QRZ^{-1} =
\begin{bmatrix}
2 & -1 \\
1 & 1 \\
0 & 3
\end{bmatrix}
\begin{bmatrix}
-2 & 3 \\
1 & -1
\end{bmatrix}^{-1}
$$
What does the LLL algorithm do?

Example ($\omega = 0.75$)

\[
\begin{bmatrix}
1 & 4 \\
2 & 5 \\
3 & 6 \\
\end{bmatrix} = QRZ^{-1} = \begin{bmatrix}
2 & -1 \\
1 & 1 \\
0 & 3 \\
\end{bmatrix} \begin{bmatrix}
-2 & 3 \\
1 & -1 \\
\end{bmatrix}^{-1}
\]

Making a lattice grid closer to orthogonal.
How may the LLL algorithm help?

Two ways:
Reduce search radius
Reduce the number of search paths
Two ways:
Reduce search radius
Reduce the number of search paths

Example

\[
A = \begin{bmatrix}
1 & 4 \\
2 & 5 \\
3 & 6 \\
\end{bmatrix} \quad b = \begin{bmatrix}
1 \\
1 \\
1 \\
\end{bmatrix}
\]

ILLS solution \( \mathbf{z} \) = \[
\begin{bmatrix}
-2 \\
1 \\
\end{bmatrix}
\]

distance \( \| A\mathbf{z} - b \|_2 \) = \( \sqrt{2} \)
Reducing search radius

QR decomposition

\[ R = \begin{bmatrix} 3.7417 & 8.5524 \\ 0 & 1.9640 \end{bmatrix} \quad \hat{b} = \begin{bmatrix} 1.6036 \\ 0.6547 \end{bmatrix} \]

LLL algorithm (\( \omega = 0.75 \))

\[ \tilde{R} = \begin{bmatrix} 2.2361 & -0.4472 \\ 0 & 3.2864 \end{bmatrix} \quad \tilde{b} = \begin{bmatrix} 1.3416 \\ 1.0955 \end{bmatrix} \]
Reducing search radius

QR decomposition

\[ R = \begin{bmatrix} 3.7417 & 8.5524 \\ 0 & 1.9640 \end{bmatrix} \quad \hat{b} = \begin{bmatrix} 1.6036 \\ 0.6547 \end{bmatrix} \]

LLL algorithm (\( \omega = 0.75 \))

\[ \tilde{R} = \begin{bmatrix} 2.2361 & -0.4472 \\ 0 & 3.2864 \end{bmatrix} \quad \tilde{b} = \begin{bmatrix} 1.3416 \\ 1.0955 \end{bmatrix} \]

Suppose we use

\[ \rho = \| R \left( R^{-1} \hat{b} \right) - \hat{b} \|_2 \]

\[ \tilde{\rho} = \| \tilde{R} \left( R^{-1} \tilde{b} \right) - \tilde{b} \|_2 \]

as the search radii, then

\[ \rho = 1.7321 \quad \text{and} \quad \tilde{\rho} = 1.4142 \]
Reducing the number of search paths

\[ R = \begin{bmatrix} 3.7417 & 8.5524 \\ 0 & 1.9640 \end{bmatrix}, \quad \hat{b} = \begin{bmatrix} 1.6036 \\ 0.6547 \end{bmatrix} \]

There are two integers \( x_2 = 0, 1 \) satisfying

\[ |r_{2,2}x_2 - \hat{b}_2| \leq \rho (1.7321) \]
Reducing the number of search paths

\[ R = \begin{bmatrix} 3.7417 & 8.5524 \\ 0 & 1.9640 \end{bmatrix} \quad \hat{b} = \begin{bmatrix} 1.6036 \\ 0.6547 \end{bmatrix} \]

There are two integers \( x_2 = 0, 1 \) satisfying

\[ |r_{2,2}x_2 - \hat{b}_2| \leq \rho (1.7321) \]

\[ \tilde{R} = \begin{bmatrix} 2.2361 & -0.4472 \\ 0 & 3.2864 \end{bmatrix} \quad \tilde{b} = \begin{bmatrix} 1.3416 \\ 1.0955 \end{bmatrix} \]

There is one integer \( x_2 = 0 \) satisfying

\[ |\tilde{r}_{2,2}x_2 - \tilde{b}_2| \leq \tilde{\rho} (1.4142) \]
Reducing the number of search paths

\[ R = \begin{bmatrix} 3.7417 & 8.5524 \\ 0 & 1.9640 \end{bmatrix} \quad \hat{b} = \begin{bmatrix} 1.6036 \\ 0.6547 \end{bmatrix} \]

There are two integers \( x_2 = 0, 1 \) satisfying

\[ |r_{2,2} x_2 - \hat{b}_2| \leq \rho \ (1.7321) \]

\[ \tilde{R} = \begin{bmatrix} 2.2361 & -0.4472 \\ 0 & 3.2864 \end{bmatrix} \quad \tilde{b} = \begin{bmatrix} 1.3416 \\ 1.0955 \end{bmatrix} \]

There is one integer \( x_2 = 0 \) satisfying

\[ |\tilde{r}_{2,2} x_2 - \tilde{b}_2| \leq \tilde{\rho} \ (1.4142) \]

Even if we use 1.7321 as the radius here, there is still one integer 0.
Search trees

$\tilde{Q} \tilde{R} = RZ, \quad Z = \begin{bmatrix} -2 & 3 \\ 1 & -1 \end{bmatrix}$

$\begin{bmatrix} -2 \\ 1 \end{bmatrix} = Z \begin{bmatrix} 1 \\ 0 \end{bmatrix}$
Search trees

\[ \tilde{Q}\tilde{R} = RZ, \quad Z = \begin{bmatrix} -2 & 3 \\ 1 & -1 \end{bmatrix} \]

\[ \begin{bmatrix} -2 \\ 1 \end{bmatrix} = Z \begin{bmatrix} 1 \\ 0 \end{bmatrix} \]

Reducing the number of search paths in the early stages of a DFS can significantly reduce the total number of search paths.
Outline

1. Introduction
   - Application
   - Integer Least Squares

2. Sphere Decoding
   - Reducing Dimension
   - Searching Lattice Points
   - Choosing a Radius

3. The LLL Algorithm

4. Conclusion
Our preliminary experiments show:
The combination of our technique for choosing search radius and the LLL algorithm can reduce running time by almost 50%.
Our preliminary experiments show:
The combination of our technique for choosing search radius and the LLL algorithm can reduce running time by almost 50%.

Future work
- complex
- consider computational error in calculating search radius
- extensive experiments on various $A$ and $b$ to investigate numerical behavior
Thank you!
Thank you!

Questions?