Algebraic decoding of the Golden Code

LAURA LUZZI

joint work with G. Rekaya-Ben Othman and J.-C. Belfiore

"THE ARITHMETICS OF WIRELESS COMMUNICATIONS"
PISA, 17 NOV 2008

Outline

Outline

MIMO systems

- The use of **multiple antennas** allows for increased data rates and reliability.
- **Algebraic number theory** is an effective tool to design codes that are full-rate and information-lossless.
- In order to increase data rates, both the number of antennas and the size of the signal set can be increased.
- This entails a high **decoding complexity** with is a real challenge for practical implementation.

System model

m transmit antennas, n receive antennas, t code length

$$\underset{\text{received signal}}{\mathbf{Y}_{n\times t}} = \underset{\text{channel}}{\mathbf{H}_{n\times m}} \; \underset{\text{codeword}}{\mathbf{X}_{m\times t}} + \underset{\text{noise}}{\mathbf{N}_{n\times t}}$$

System model

m transmit antennas, n receive antennas, t code length

$$\underbrace{\mathbf{Y}_{n\times t}}_{\text{received signal}} = \underbrace{\mathbf{H}_{n\times m}}_{\text{channel}} \underbrace{\mathbf{X}_{m\times t}}_{\text{codeword}} + \underbrace{\mathbf{N}_{n\times t}}_{\text{noise}}$$

- in our model, $n = m = t$
- $\mathcal{A}$ **division algebra** of degree n^2 over $\mathbb{Q}(i)$
- $\mathcal{O}$ **maximal order** ($\mathbb{Z}[i]$-lattice) of $\mathcal{A}$, $\mathcal{O}\alpha$ **ideal** of $\mathcal{O}$
- $\mathbf{X} \in \mathcal{O}\alpha$

System model

m transmit antennas, n receive antennas, t code length

$$\underset{\text{received signal}}{\mathbf{Y}_{n\times t}} = \underset{\text{channel}}{\mathbf{H}_{n\times m}} \; \underset{\text{codeword}}{\mathbf{X}_{m\times t}} + \underset{\text{noise}}{\mathbf{N}_{n\times t}}$$

- in our model, $n = m = t$
- $\mathcal{A}$ **division algebra** of degree n^2 over $\mathbb{Q}(i)$
- $\mathcal{O}$ **maximal order** ($\mathbb{Z}[i]$-lattice) of $\mathcal{A}$, $\quad \mathcal{O}\alpha$ **ideal** of $\mathcal{O}$
- $\mathbf{X} \in \mathcal{O}\alpha$

Golden Code: $\quad n = 2, \quad (s_1, s_2, s_3, s_4) \in \mathbb{Z}[i]^4$ QAM symbols

$$\mathbf{X} = \frac{1}{\sqrt{5}} \begin{pmatrix} \alpha(s_1 + s_2\theta) & \alpha(s_3 + s_4\theta) \\ \bar{\alpha} i(s_3 + s_4\bar{\theta}) & \bar{\alpha}(s_1 + s_2\bar{\theta}) \end{pmatrix}, \quad \theta \text{ golden number}$$

Lattice point representation

Matrix form: $\{\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{W}_4\}$ basis of $\alpha\mathcal{O}$ as a $\mathbb{Z}[i]$-module.

$\mathbf{X} = \sum s_i \mathbf{W}_i$, $\mathbf{s} = (s_1, s_2, s_3, s_4) \in \mathbb{Z}[i]^4$ vector of QAM information signals

Vector form: $\mathbf{x} = \sum s_i \mathbf{w}_i = \Phi \mathbf{s}$

$$\mathbf{y} = \mathbf{H}_l \Phi \mathbf{s} + \mathbf{n}$$

- $\mathbf{H}_l$ linear map corresponding to left multiplication by $\mathbf{H}$
- Φ generator matrix of the code lattice with columns $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, \mathbf{w}_4$

ML decoding: $\hat{\mathbf{s}} = \operatorname*{argmin}_{\mathbf{s} \in \text{QAM}} \|\mathbf{y} - \mathbf{H}_l \Phi \mathbf{s}\|$

Decoding

Up to now, decoding has been performed using the **lattice point representation**

- **ML decoders** (Sphere Decoder, Schnorr-Euchner...)
  - optimal performance but with high complexity
- **Suboptimal decoders** (Zero-forcing, MMSE...)
  - reduced complexity but with poor performance

Decoding

Up to now, decoding has been performed using the **lattice point representation**

- **ML decoders** (Sphere Decoder, Schnorr-Euchner...)
  - optimal performance but with high complexity
- **Suboptimal decoders** (Zero-forcing, MMSE...)
  - reduced complexity but with poor performance

- The use of **preprocessing** before decoding can reduce the complexity of ML decoders and improve the performance of suboptimal decoders.
  - **left preprocessing** (MMSE-GDFE) to obtain a better conditioned channel matrix
  - **right preprocessing** (lattice reduction) to have a quasi-orthogonal lattice

Algebraic reduction

- Up to now, algebraic tools have been used for coding but not for decoding
- Algebraic reduction is a right preprocessing method that exploits the ring structure of the code

Principle: Part of the channel is absorbed by the code

- Approximate the channel matrix with a unit of the maximal order $\mathcal{O}$
- The approximation error should be quasi-unitary

Outline

The SISO case: system model

$$\underset{\text{received signal}}{\mathbf{y}} = \underset{\text{channel}}{\mathbf{H}} \underset{\text{codeword}}{\mathbf{x}} + \underset{\text{noise}}{\mathbf{n}}$$

- **H** is **diagonal**
- K cyclotomic extension of $\mathbb{Q}(i)$ of degree n, $\quad \mathrm{Gal}(K/\mathbb{Q}(i)) =< \sigma >$
- $\mathcal{O}_K$ ring of integers of K, $\quad \{w_1, \ldots, w_n\}$ basis of $\mathcal{O}_K$ over $\mathbb{Z}[i]$.
- **canonical embedding** $\quad \mathcal{O}_K \to \mathbb{C}^n$

$$x \mapsto \mathbf{x} = (x, \sigma(x), \ldots, \sigma^{n-1}(x))^t$$

- $x = s_1 w_1 + \ldots + s_n w_n \in \mathcal{O}_K$, $\quad \mathbf{x} = \Phi \mathbf{s}$ codeword, $\quad \mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{Z}[i]^n$

Algebraic reduction for fast fading channels

[Rekaya, Belfiore, Viterbo 2004]

- Normalization of the received signal: $\mathbf{y}' = \frac{\mathbf{y}}{\sqrt[n]{\det(\mathbf{H})}} = \mathbf{H_1 x} + \mathbf{n}'$
- **Principle:** approximate $\mathbf{H_1} = \operatorname{diag}(h_1, \ldots, h_n)$ with $\mathbf{U} = \operatorname{diag}(u, \sigma(u), \ldots, \sigma^{n-1}(u))$, where u is a **unit** of $\mathcal{O}_K$.

u unit of $\mathcal{O}_K$ $\Leftrightarrow$ $\mathbf{U}\Phi = \Phi\mathbf{T}$ with $\mathbf{T}$ **unimodular** (with entries in $\mathbb{Z}[i]$).

$$\mathbf{y}' \sim \mathbf{U}\Phi\mathbf{s} + \mathbf{n}' = \Phi\mathbf{T}\mathbf{s} + \mathbf{n}' = \Phi\mathbf{s}' + \mathbf{n}', \qquad \mathbf{s}' \in \mathbb{Z}[i]^n$$

- Φ unitary $\Rightarrow$ ZF decoding is quasi-optimal
- **How to find U?**

The logarithmic lattice

Dirichlet unit theorem

$$u = u_1^{n_1} \cdots u_k^{n_k}, \qquad u_1, \ldots, u_k \quad \text{fundamental units.}$$

- $\mathbf{u}$ canonical embedding of u.

$$\log |\mathbf{u}| = n_1 \log |\mathbf{u_1}| + \ldots + n_k \log |\mathbf{u_k}|$$

  belongs to the **logarithmic lattice** generated by $\{\log |\mathbf{u_1}|, \ldots, \log |\mathbf{u_1}|\}$.

- **Solution:** find the closest point to $(\log |h_1|, \ldots, \log |h_n|)$ in the logarithmic lattice.

- **Advantage:** this lattice is fixed once and for all and doesn't depend on the channel.

Outline

Outline

Perfect approximation

Normalization of the received signal: $\mathbf{Y}' = \frac{\mathbf{Y}}{\sqrt{\det(\mathbf{H})}}$

$$\mathbf{Y}' = \mathbf{H_1}\mathbf{X} + \mathbf{N}', \quad \det(\mathbf{H_1}) = 1$$

Ideal case

- Suppose that $\mathbf{H_1}$ is a unit $\mathbf{U}$ of $\mathcal{O}$: $\mathbf{Y}' = \mathbf{U}\mathbf{X} + \mathbf{N}'$
- $\mathbf{U}\mathbf{X} = \mathbf{X}'$ is still a codeword:

$$\{\mathbf{U}\mathbf{X} \mid \mathbf{X} \in \mathcal{O}\alpha\} = \mathcal{O}\alpha$$

- It is equivalent to a non-fading channel $\mathbf{Y}' = \mathbf{X}' + \mathbf{N}'$

Perfect approximation

In vectorized form:

$$\mathbf{y}' = \mathbf{U}_l \Phi \mathbf{s} + \mathbf{n}'$$

- $\mathbf{U}_l$ linear map corresponding to left multiplication by $\mathbf{U}$
- Φ generator matrix of the code lattice
- $\mathbf{s} \in \mathbb{Z}[i]^4$ vector of QAM information signals

U unit $\quad \Leftrightarrow \quad$ $\mathbf{U}_l \Phi = \Phi \mathbf{T}$ **with T unimodular**

$$\mathbf{y}' = \Phi \mathbf{T} \mathbf{s} + \mathbf{n}' = \Phi \mathbf{s}' + \mathbf{n}' \qquad \mathbf{s}' \in \mathbb{Z}[i]^4$$

General case

In general the approximation is not perfect:

$$\mathbf{H}_1 = \mathbf{E}\mathbf{U}, \qquad \mathbf{E} \quad \text{approximation error}$$

General case

In general the approximation is not perfect:

$$\mathbf{H}_1 = \mathbf{E}\mathbf{U}, \qquad \mathbf{E} \quad \text{approximation error}$$

Perfect approximation

$$\mathbf{y}' = \mathbf{U}_l \Phi \mathbf{s} + \mathbf{n}' =$$
$$= \Phi \mathbf{T}\mathbf{s} + \mathbf{n}'$$

Φ unitary

$\Rightarrow$ **ZF decoding is optimal**

$$\mathbf{s}' = \mathbf{T}\mathbf{s} = \left[\Phi^{-1}\mathbf{y}'\right]$$

General case

In general the approximation is not perfect:

$$\mathbf{H}_1 = \mathbf{E}\mathbf{U}, \qquad \mathbf{E} \quad \text{approximation error}$$

Perfect approximation

$$\mathbf{y}' = \mathbf{U}_l \Phi \mathbf{s} + \mathbf{n}' =$$
$$= \Phi \mathbf{T}\mathbf{s} + \mathbf{n}'$$

Φ unitary

$\Rightarrow$ **ZF decoding is optimal**

$$\mathbf{s}' = \mathbf{T}\mathbf{s} = \left[\Phi^{-1}\mathbf{y}'\right]$$

General case

$$\mathbf{y}' = \mathbf{E}_l \mathbf{U}_l \Phi \mathbf{s} + \mathbf{n}' =$$
$$= \mathbf{E}_l \Phi \mathbf{T}\mathbf{s} + \mathbf{n}'$$

To have quasi-optimal decoding
$\mathbf{E}$ must be quasi-unitary

$\Rightarrow$ **Choose U such that $\|\mathbf{E}\|_F$ is minimized**

Outline

How to find $\mathbf{U}$?

The group of units of $\mathcal{O}$ is a **discrete subgroup** Γ **of** $\mathrm{SL}_2(\mathbb{C})$.

How to find $\mathbf{U}$?

The group of units of $\mathcal{O}$ is a **discrete subgroup** Γ **of** $SL_2(\mathbb{C})$.

Problem:

$$\mathbf{H}_1 \in SL_2(\mathbb{C}) \quad \longrightarrow \quad \text{find } \mathbf{U} \in \Gamma \text{ s.t. } \|\mathbf{E}\|_F = \left\|\mathbf{H_1 U^{-1}}\right\|_F \text{ is small}$$

How to find U?

The group of units of $\mathcal{O}$ is a **discrete subgroup** Γ **of** $SL_2(\mathbb{C})$.

Problem:

$$\mathbf{H}_1 \in SL_2(\mathbb{C}) \quad \longrightarrow \quad \text{find } \mathbf{U} \in \Gamma \text{ s.t. } \|\mathbf{E}\|_F = \left\|\mathbf{H_1 U^{-1}}\right\|_F \text{ is small}$$

Action of $SL_2(\mathbb{C})$ on **hyperbolic** 3**-space** $\mathbb{H}^3$

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$J = (0,0,1) \quad \mapsto \quad A(J) = \left(\frac{\mathrm{Re}(b\bar{d} + a\bar{c})}{|c|^2 + |d|^2}, \frac{\mathrm{Im}(b\bar{d} + a\bar{c})}{|c|^2 + |d|^2}, \frac{1}{|c|^2 + |d|^2} \right)$$

How to find $\mathbf{U}$?

The group of units of $\mathcal{O}$ is a **discrete subgroup** Γ **of** $SL_2(\mathbb{C})$.

Problem:

$$\mathbf{H}_1 \in SL_2(\mathbb{C}) \quad \longrightarrow \quad \text{find } \mathbf{U} \in \Gamma \text{ s.t. } \|\mathbf{E}\|_F = \left\|\mathbf{H_1 U^{-1}}\right\|_F \text{ is small}$$

Action of $SL_2(\mathbb{C})$ on **hyperbolic 3-space** $\mathbb{H}^3$

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

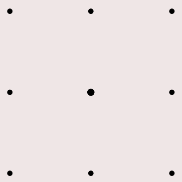
$$J = (0,0,1) \quad \mapsto \quad A(J) = \left(\frac{\mathrm{Re}(b\bar{d} + a\bar{c})}{|c|^2 + |d|^2}, \frac{\mathrm{Im}(b\bar{d} + a\bar{c})}{|c|^2 + |d|^2}, \frac{1}{|c|^2 + |d|^2} \right)$$

$\left\|\mathbf{H_1 U^{-1}}\right\|_F$ **is small** $\quad \Leftrightarrow \quad \mathbf{U}^{-1}(J)$ **is close to** $\mathbf{H_1}^{-1}(J)$ in hyperbolic distance

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



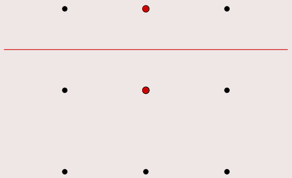
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



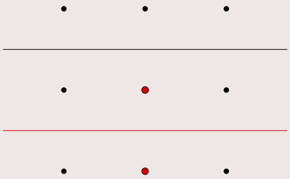
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



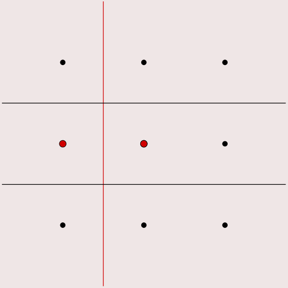
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



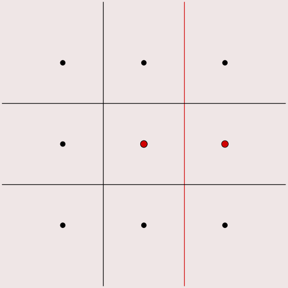
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



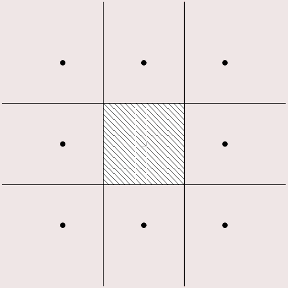
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

- the area enclosed by **bisectors** is a **fundamental domain** for the action



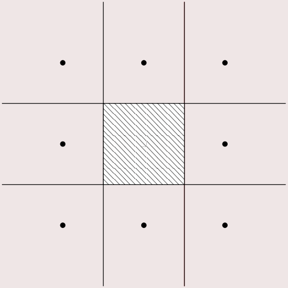
- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

Action of Γ on $\mathbb{H}^3$

Discrete subgroups and fundamental domains

Example: action of $\mathbb{Z}^2$ on $\mathbb{R}^2$

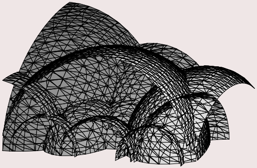
- the area enclosed by **bisectors** is a **fundamental domain** for the action



- the images of the fundamental domain form a **tiling** of $\mathbb{R}^2$

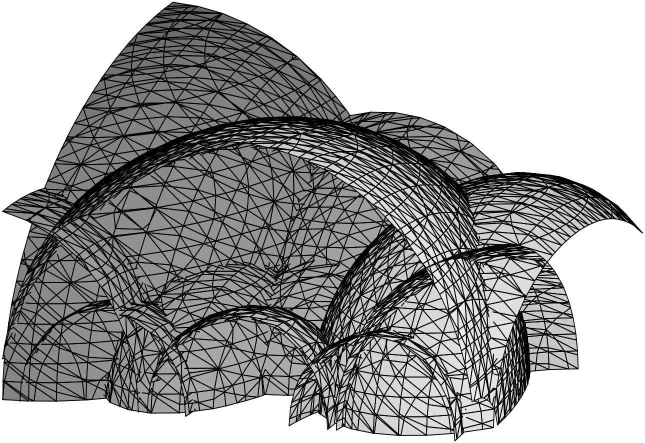
Action of Γ on $\mathbb{H}^3$

- the bisectors are Euclidean spheres



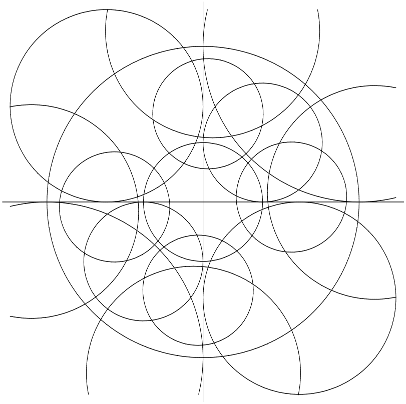
- the fundamental domain is a **hyperbolic polyhedron**
- the images of the fundamental domain form a tiling of $\mathbb{H}^3$

Intersecting bisectors

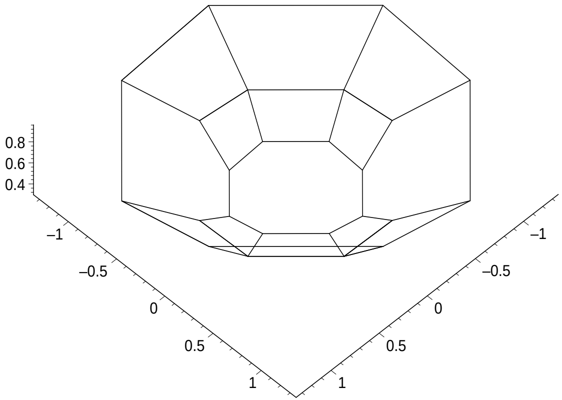


Intersecting bisectors

Projection on the plane $\{z = 0\}$

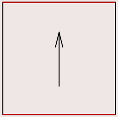


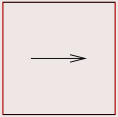
The fundamental polyhedron



Finding the generators

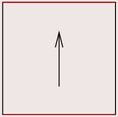
- The **generators** of the group correspond to the **side-pairings** of the fundamental polyhedron

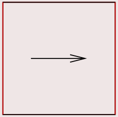




Finding the generators

- The **generators** of the group correspond to the **side-pairings** of the fundamental polyhedron





Golden Code: 8 generators for the unit group

$$U_1 = \begin{pmatrix} i\theta & 0 \\ 0 & i\bar{\theta} \end{pmatrix} \qquad U_5 = \begin{pmatrix} 1+i & 1+i\bar{\theta} \\ i(1+i\theta) & 1+i \end{pmatrix}$$

$$U_2 = \begin{pmatrix} i & 1+i \\ i-1 & i \end{pmatrix} \qquad U_6 = \begin{pmatrix} 1+i & 1+i\theta \\ i(1+i\bar{\theta}) & 1+i \end{pmatrix}$$

$$U_3 = \begin{pmatrix} \theta & 1+i \\ i-1 & \bar{\theta} \end{pmatrix} \qquad U_7 = \begin{pmatrix} 1-i & \bar{\theta}+i \\ i(\theta+i) & 1-i \end{pmatrix}$$

$$U_4 = \begin{pmatrix} \theta & -1-i \\ -i+1 & \bar{\theta} \end{pmatrix} \qquad U_8 = \begin{pmatrix} 1-i & \theta+i \\ i(\bar{\theta}+i) & 1-i \end{pmatrix}$$

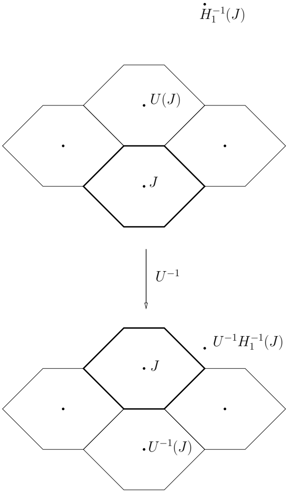
The algorithm

- the polyhedra adjacent to the fundamental polyhedron $\mathcal{P}$ are of the form $\mathbf{U}(\mathcal{P})$, with $\mathbf{U}$ a generator

Unit search algorithm

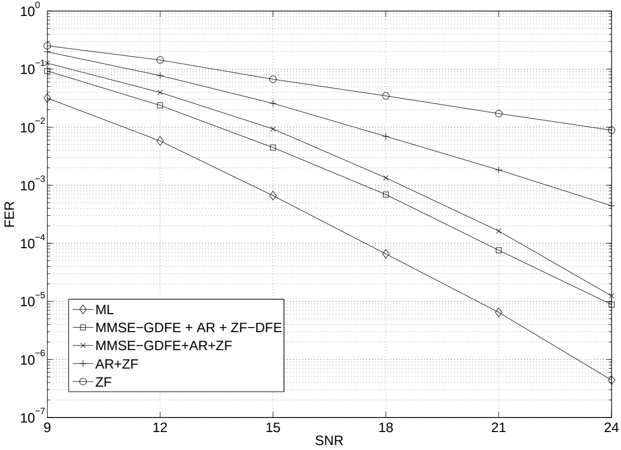
1) find the generator $\mathbf{U}$ such that $\mathbf{U}(J)$ is closest to $\mathbf{H_1}^{-1}(J)$
2) every $\mathbf{U}$ is an isometry $\Rightarrow$ apply $\mathbf{U}^{-1}$

- Repeat steps 1-2 until J is the closest point to $\mathbf{H_1}^{-1}(J)$

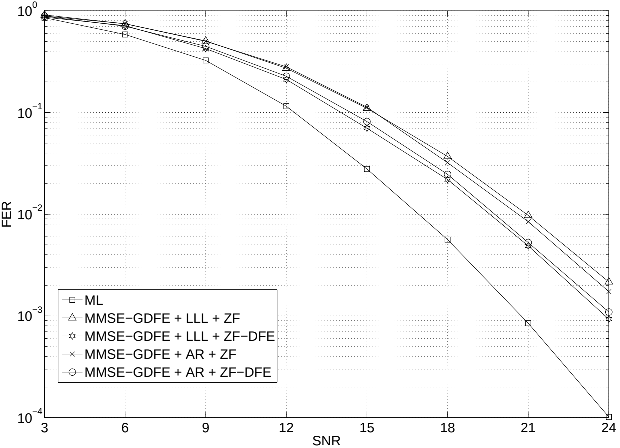


Outline

Performance of the algebraic reduction - 4-QAM



Comparison with LLL reduction - 16-QAM



Outline

Conclusion and perspectives

Conclusion

- **performance:** algebraic reduction is at 3.4 dB from ML performance using MMSE-GDFE preprocessing and ZF decoding
- **advantage over LLL reduction:** for slow-fading channels, the search algorithm only requires a small update at each step instead of a full reduction

Open problems

- find good codes such that the group of units has the **smallest possible number of generators**
- extend algebraic reduction to **higher-dimensional** space-time codes based on division algebras (e.g. Perfect Codes)