

# Lattice Basis Reduction

## Part II: Algorithms

Sanzheng Qiao

Department of Computing and Software  
McMaster University, Canada  
qiao@mcmaster.ca  
www.cas.mcmaster.ca/~qiao

November 8, 2011, revised February 2012

Joint work with W. Zhang and Y. Wei, Fudan University

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction
- 3 HKZ Reduction
- 4 Minkowski Reduction
- 5 A Measurement

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction
- 3 HKZ Reduction
- 4 Minkowski Reduction
- 5 A Measurement

# Hermite reduction (size reduction)

## Hermite-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called size-reduced if its QR decomposition satisfies

$$|r_{i,i}| \geq 2|r_{i,j}|, \quad \text{for all } 1 \leq i < j \leq n,$$

# Hermite reduction (size reduction)

## Hermite-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called size-reduced if its QR decomposition satisfies

$$|r_{i,i}| \geq 2|r_{i,j}|, \quad \text{for all } 1 \leq i < j \leq n,$$

Procedure Reduce( $i, j$ )

$$\begin{bmatrix} r_{i,i} & r_{i,j} \\ & r_{j,j} \end{bmatrix} \begin{bmatrix} 1 & -\left\lfloor \frac{r_{i,j}}{r_{i,i}} \right\rfloor \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{i,i} & r_{i,j} - r_{i,i} \left\lfloor \frac{r_{i,j}}{r_{i,i}} \right\rfloor \\ & r_{j,j} \end{bmatrix}$$

$$|r_{i,i}| \geq 2 \left| r_{i,j} - r_{i,i} \left\lfloor \frac{r_{i,j}}{r_{i,i}} \right\rfloor \right|$$

# Gauss reduction

A unimodular transformation

$$\begin{bmatrix} 1 & -\mu \\ 0 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 0 \\ -\mu & 1 \end{bmatrix}$$

Also called

Integer Gauss transformation

Integer elementary matrix

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction**
- 3 HKZ Reduction
- 4 Minkowski Reduction
- 5 A Measurement

# LLL reduction

## LLL-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called LLL-reduced if it is size-reduced and  $R$  in the QR decomposition satisfies

$$r_{i+1,i+1}^2 + r_{i,i+1}^2 \geq \omega r_{i,i}^2$$

# LLL reduction

## LLL-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called LLL-reduced if it is size-reduced and  $R$  in the QR decomposition satisfies

$$r_{i+1,i+1}^2 + r_{i,i+1}^2 \geq \omega r_{i,i}^2$$

Procedure `SwapRestore( $i$ )`

Find a Givens plane rotation  $G$ :

$$G \begin{bmatrix} r_{i-1,i-1} & r_{i-1,i} \\ 0 & r_{i,i} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \hat{r}_{i-1,i-1} & \hat{r}_{i-1,i} \\ 0 & \hat{r}_{i,i} \end{bmatrix}.$$

Unimodular transformation: Permutation

# LLL algorithm

```
k = 2;
while k <= n {
  if |r(k-1,k) / r(k-1,k-1)| > 1/2

  if r(k,k)^2 + r(k-1,k)^2 < w*r(k-1,k-1)^2 {

  } else {

  }
}
```

# LLL algorithm

```
k = 2;
while k <= n {
  if |r(k-1,k) / r(k-1,k-1)| > 1/2
    Reduce(k-1,k);
  if r(k,k)^2 + r(k-1,k)^2 < w*r(k-1,k-1)^2 {

  } else {

  }
}
```

# LLL algorithm

```
k = 2;
while k <= n {
  if |r(k-1,k) / r(k-1,k-1)| > 1/2
    Reduce(k-1,k);
  if r(k,k)^2 + r(k-1,k)^2 < w*r(k-1,k-1)^2 {
    SwapRestore(k);
    k = max(k-1, 2);
  } else {
    }
}
```

# LLL algorithm

```
k = 2;
while k <= n {
  if |r(k-1,k) / r(k-1,k-1)| > 1/2
    Reduce(k-1,k);
  if r(k,k)^2 + r(k-1,k)^2 < w*r(k-1,k-1)^2 {
    SwapRestore(k);
    k = max(k-1, 2);
  } else {
    for i = k-2 downto 1
      if |r(i,k) / r(i,i)| > 1/2
        Reduce(i,k);
    k = k+1;
  }
}
```

# LLL algorithm

```
k = 2;
while k <= n {
  if |r(k-1,k) / r(k-1,k-1)| > 1/2
    Reduce(k-1,k);
  if r(k,k)^2 + r(k-1,k)^2 < w*r(k-1,k-1)^2 {
    SwapRestore(k);
    k = max(k-1, 2);
  } else {
    for i = k-2 downto 1
      if |r(i,k) / r(i,i)| > 1/2
        Reduce(i,k);
    k = k+1;
  }
}
```

Redundant size reductions.

# An improvement: Delayed size reduction

```
k = 2;
while k <= n
  g = round(r(k-1,k) / r(k-1,k-1));
  if r(k,k)^2 + (r(k-1,k) - g*r(k-1,k-1))^2 <
      w*r(k-1,k-1)^2
    ReduceSwapRestore(k);
    k = max(k-1, 2);
  else
    k = k + 1;

for k = 2 to n
  for i = k-1 downto 1
    if |r(i,k) / r(i,i)| > 1/2
      Reduce(i,k);
```

# An improvement: Delayed size reduction

```
k = 2;
while k <= n
  g = round(r(k-1,k) / r(k-1,k-1));
  if r(k,k)^2 + (r(k-1,k) - g*r(k-1,k-1))^2 <
      w*r(k-1,k-1)^2
    ReduceSwapRestore(k);
    k = max(k-1, 2);
  else
    k = k + 1;

for k = 2 to n
  for i = k-1 downto 1
    if |r(i,k) / r(i,i)| > 1/2
      Reduce(i,k);
```

Produces identical results at 50% cost.

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction
- 3 HKZ Reduction**
- 4 Minkowski Reduction
- 5 A Measurement

# HKZ reduction

## HKZ-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called HKZ-reduced if it is size-reduced and for each trailing  $(n - i + 1) \times (n - i + 1)$ ,  $1 \leq i < n$ , submatrix of  $R$  in the QR decomposition, its first column is a shortest nonzero vector in the lattice generated by the submatrix.

# HKZ reduction

## HKZ-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called HKZ-reduced if it is size-reduced and for each trailing  $(n - i + 1) \times (n - i + 1)$ ,  $1 \leq i < n$ , submatrix of  $R$  in the QR decomposition, its first column is a shortest nonzero vector in the lattice generated by the submatrix.

## Two problems

- Shortest vector problem (SVP)
- Expansion to a basis

## SVP

$$\min_z \|Bz\|_2^2$$

# SVP

$$\min_z \|Bz\|_2^2$$

Sphere decoding

Determine a search sphere

$$\|Bz\|_2^2 \leq \rho^2$$

## SVP

$$\min_z \|Bz\|_2^2$$

Sphere decoding

Determine a search sphere

$$\|Bz\|_2^2 \leq \rho^2$$

A simple choice of  $\rho$ : the length of the first (or shortest) column of  $B$ .

# Example

$$Rz = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$$

$$\rho = 4$$

# Example

$$Rz = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$$

$$\rho = 4$$

A necessary condition for  $z_3$ :  $|3z_3| \leq 4$ .

Possible values of  $z_3$ : 0, -1, 1

# Example

For each possible values of  $z_3$ , say  $z_3 = 0$ ,

$$Rz = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} 4 & 1 \\ 0 & 4 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + 0 \begin{bmatrix} 5 \\ 4 \\ 3 \end{bmatrix}$$

The problem size is reduced.

# Example

For each possible values of  $z_3$ , say  $z_3 = 0$ ,

$$Rz = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} 4 & 1 \\ 0 & 4 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + 0 \begin{bmatrix} 5 \\ 4 \\ 3 \end{bmatrix}$$

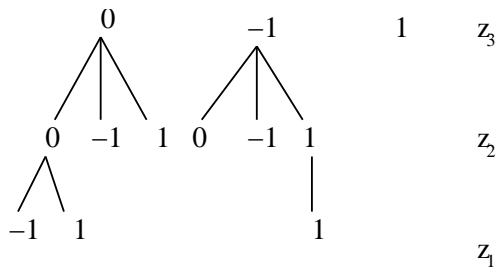
The problem size is reduced.

The necessary condition for  $z_2$ :  $|4z_2| \leq 4$

Possible values of  $z_2$ : 0, -1, 1

# Example

The search tree



The solution

$$Rz = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -3 \end{bmatrix}$$

# Expanding to a basis

Problem:

Transform the basis matrix

$$A = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix}$$

into a new basis matrix whose first column is the shortest vector

$$Az = \begin{bmatrix} 0 \\ 0 \\ -3 \end{bmatrix}$$

# Expanding to a basis

Problem:

Transform the basis matrix

$$A = \begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix}$$

into a new basis matrix whose first column is the shortest vector

$$Az = \begin{bmatrix} 0 \\ 0 \\ -3 \end{bmatrix}$$

That is, find a unimodular matrix  $Z$ :  $Az = AZ\mathbf{e}_1$  or

$$\mathbf{z} = Z\mathbf{e}_1, \quad Z^{-1}\mathbf{z} = \mathbf{e}_1$$

Unimodular transformation that introduces zeros into an integer vector.

# A plane unimodular transformation

A unimodular transformation (Luk, Zhang, and Q, 2010).

$$\gcd(p, q) = \pm d, \quad ap + bq = \pm d.$$

Form the unimodular matrix

$$\begin{bmatrix} a & b \\ -q/d & p/d \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}$$

# A plane unimodular transformation

A unimodular transformation (Luk, Zhang, and Q, 2010).

$$\gcd(p, q) = \pm d, \quad ap + bq = \pm d.$$

Form the unimodular matrix

$$\begin{bmatrix} a & b \\ -q/d & p/d \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}$$

Its inverse

$$\begin{bmatrix} p/d & -b \\ q/d & a \end{bmatrix}$$

# Example

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 4 & 1 & 5 \\ 0 & 4 & 4 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & -4 & 5 \\ 0 & 0 & 4 \\ 0 & -3 & 3 \end{bmatrix}$$

# Example

$$\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 4 & -4 & 5 \\ 0 & 0 & 4 \\ 0 & -3 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -4 & 5 \\ 0 & 0 & 4 \\ -3 & -3 & 3 \end{bmatrix}$$

# Improving Kannan's algorithm

Kannan, 1987

Expansion method

In the  $k$ th,  $k = 1, \dots, n$ , recursion, solve a  $k$ -dim system ( $O(k^3)$ ).

Total  $O(n^4)$

# Improving Kannan's algorithm

Kannan, 1987

Expansion method

In the  $k$ th,  $k = 1, \dots, n$ , recursion, solve a  $k$ -dim system ( $O(k^3)$ ).

Total  $O(n^4)$

Determine whether a set of vectors are linearly dependent.

# Improving Kannan's algorithm

Kannan, 1987

Expansion method

In the  $k$ th,  $k = 1, \dots, n$ , recursion, solve a  $k$ -dim system ( $O(k^3)$ ).

Total  $O(n^4)$

Determine whether a set of vectors are linearly dependent.

Our method

Efficient,  $O(n^2)$

Accurate, unimodular (integer) transformations.

# Properties

- Efficient.
- Exact, integer arithmetic.
- Include permutation and identity as special cases.
- Can triangularize an integer matrix.
- Any unimodular can be decomposed into a product of this plan unimodular and integer Gauss transformations.

# Application

Cryptography

Find a large vector

$$v = \begin{bmatrix} 997 \\ 1234 \\ 56789 \end{bmatrix}, \quad \gcd(v_i) = 1$$

# Application

Cryptography

Find a large vector

$$v = \begin{bmatrix} 997 \\ 1234 \\ 56789 \end{bmatrix}, \quad \gcd(v_i) = 1$$

Determine a unimodular matrix

$$Z^{-1}v = \begin{bmatrix} 997 & -1 & 0 \\ 1234 & 0 & -543 \\ 6789 & 0 & -24989 \end{bmatrix}^{-1} v = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{cond}(Z) = 1.55 \times 10^{12}$$

# Application

Cryptography

Find a large vector

$$v = \begin{bmatrix} 997 \\ 1234 \\ 56789 \end{bmatrix}, \quad \gcd(v_i) = 1$$

Determine a unimodular matrix

$$Z^{-1}v = \begin{bmatrix} 997 & -1 & 0 \\ 1234 & 0 & -543 \\ 6789 & 0 & -24989 \end{bmatrix}^{-1} v = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{cond}(Z) = 1.55 \times 10^{12}$$

Choose a diagonal  $A$  as a private key

$B = AZ$  (ill-conditioned) as public key

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction
- 3 HKZ Reduction
- 4 Minkowski Reduction**
- 5 A Measurement

# Minkowski reduction

## Minkowski-reduced

A lattice basis  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n\}$  is called Minkowski-reduced if for each  $\mathbf{b}_i$ ,  $i = 1, 2, \dots, n$ , its length

$$\|\mathbf{b}_i\|_2 = \min(\|\hat{\mathbf{b}}_i\|_2, \|\hat{\mathbf{b}}_{i+1}\|_2, \dots, \|\hat{\mathbf{b}}_n\|_2)$$

over all sets  $\{\hat{\mathbf{b}}_i, \hat{\mathbf{b}}_{i+1}, \dots, \hat{\mathbf{b}}_n\}$  of lattice points such that  $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{i-1}, \hat{\mathbf{b}}_i, \hat{\mathbf{b}}_{i+1}, \dots, \hat{\mathbf{b}}_n\}$  form a basis for the lattice.

# Existing Minkowski reduction algorithms

- Lagrange, 1773, dimension two
- Semaev, 2001, dimension three
- Nguyen and Stehle, 2009, dimension four
- Afflerbach and Grothe, 1985, up to dimension seven
- Helfrich, 1985, theoretical value, very expensive

# Existing Minkowski reduction algorithms

- Lagrange, 1773, dimension two
- Semaev, 2001, dimension three
- Nguyen and Stehle, 2009, dimension four
- Afflerbach and Grothe, 1985, up to dimension seven
- Helfrigh, 1985, theoretical value, very expensive

Zhang, Q, Wei, 2011

# Problem

For  $p = 1, 2, \dots, n$ , find  $\mathbf{b}_p$ : a shortest vector such that  $\{\mathbf{b}_1, \dots, \mathbf{b}_p\}$  can be extended to a basis for the lattice.

# Problem

For  $p = 1, 2, \dots, n$ , find  $\mathbf{b}_p$ : a shortest vector such that  $\{\mathbf{b}_1, \dots, \mathbf{b}_p\}$  can be extended to a basis for the lattice.

Algorithm:

for  $p = 1 \dots n$

    find a shortest  $\mathbf{v} = B\mathbf{z}$  such that

$\{\mathbf{b}_1, \dots, \mathbf{b}_{p-1}, \mathbf{v}\}$  is expandable to a basis;

    set  $\mathbf{b}_p = \mathbf{v}$  and expand  $\{\mathbf{b}_1, \dots, \mathbf{b}_p\}$  to a basis;

end

# Minkowski reduction algorithm

A proposition:

Let  $B = [\mathbf{b}_1, \dots, \mathbf{b}_n]$  be a generator matrix for a lattice  $L$  and a lattice vector  $\mathbf{v} = B\mathbf{z}$ , then  $\{\mathbf{b}_1, \dots, \mathbf{b}_{p-1}, \mathbf{v}\}$  is expandable to a basis for  $L$  if and only if  $\gcd(z_p, \dots, z_n) = \pm 1$ .

# Minkowski reduction algorithm

A proposition:

Let  $B = [\mathbf{b}_1, \dots, \mathbf{b}_n]$  be a generator matrix for a lattice  $L$  and a lattice vector  $\mathbf{v} = B\mathbf{z}$ , then  $\{\mathbf{b}_1, \dots, \mathbf{b}_{p-1}, \mathbf{v}\}$  is expandable to a basis for  $L$  if and only if  $\gcd(z_p, \dots, z_n) = \pm 1$ .

Constrained minimization problem:

$$\min_{\mathbf{z}} \|B\mathbf{z}\|_2 \quad \text{subject to} \quad \gcd(z_p, \dots, z_n) = \pm 1$$

# Minkowski reduction algorithm

A proposition:

Let  $B = [\mathbf{b}_1, \dots, \mathbf{b}_n]$  be a generator matrix for a lattice  $L$  and a lattice vector  $\mathbf{v} = B\mathbf{z}$ , then  $\{\mathbf{b}_1, \dots, \mathbf{b}_{p-1}, \mathbf{v}\}$  is expandable to a basis for  $L$  if and only if  $\gcd(z_p, \dots, z_n) = \pm 1$ .

Constrained minimization problem:

$$\min_{\mathbf{z}} \|B\mathbf{z}\|_2 \quad \text{subject to} \quad \gcd(z_p, \dots, z_n) = \pm 1$$

Modified sphere decoding:

While searching for short lattice vectors, enforce the condition  $\gcd(z_p, \dots, z_n) = \pm 1$ .

# Outline

- 1 Hermite Reduction
- 2 LLL Reduction
- 3 HKZ Reduction
- 4 Minkowski Reduction
- 5 A Measurement**

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

How do we measure the degree of orthogonality of a basis?

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

How do we measure the degree of orthogonality of a basis?

Usual choice: *condition number* of matrix.

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

How do we measure the degree of orthogonality of a basis?

Usual choice: *condition number* of matrix.

Consider the matrix  $\begin{bmatrix} 1 & 0 \\ 0 & 10^k \end{bmatrix}$ .

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

How do we measure the degree of orthogonality of a basis?

Usual choice: *condition number* of matrix.

Consider the matrix  $\begin{bmatrix} 1 & 0 \\ 0 & 10^k \end{bmatrix}$ .

Its condition number is  $10^k$ , but the columns are orthogonal.

# Measuring orthogonality

Lattice reduction is to transform a lattice basis into another that becomes “more orthogonal”.

How do we measure the degree of orthogonality of a basis?

Usual choice: *condition number* of matrix.

Consider the matrix  $\begin{bmatrix} 1 & 0 \\ 0 & 10^k \end{bmatrix}$ .

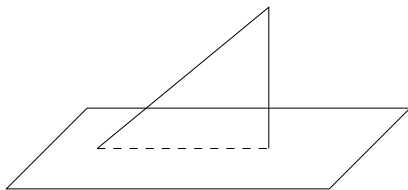
Its condition number is  $10^k$ , but the columns are orthogonal.

*Condition #* ignores intermediate singular values of  $n \times n$  matrix.

# An interpretation

$$\begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & r_{1,4} \\ 0 & r_{2,2} & r_{2,3} & r_{2,4} \\ 0 & 0 & r_{3,3} & r_{3,4} \\ 0 & 0 & 0 & r_{4,4} \end{bmatrix}$$

$$\sin \theta_j = \frac{|r_{j,i}|}{\|r_{:,i}\|_2}$$



# Measurement

In particular, the geometric mean  $\sigma$ :

$$\sigma^n = \prod_{i=1}^n \sin \theta_i = \prod_{i=1}^n \frac{|r_{i,i}|}{\|r_{:,i}\|_2} = \frac{d(L)}{\prod_{i=1}^n \|b_i\|_2}$$

Hadamard's inequality

$$\det(B) \leq \prod_{i=1}^n \|b_i\|_2$$

The equality holds if and only if  $b_i$  are orthogonal.

Also called Hadamard ratio or orthogonality defect.

# Measurement

- Note that  $0 \leq \sigma \leq 1$ ,  $\sigma = 1$  for any diagonal matrix, and  $\sigma = 0$  for any singular matrix.

# Measurement

- Note that  $0 \leq \sigma \leq 1$ ,  $\sigma = 1$  for any diagonal matrix, and  $\sigma = 0$  for any singular matrix.
- Since  $V_n = \prod_{i=1}^n |r_{i,i}| = d(L)$  is a constant for a given  $L$ , we can improve  $\sigma$  by reducing  $\|b_i\|_2$ .

# Measurement

- Note that  $0 \leq \sigma \leq 1$ ,  $\sigma = 1$  for any diagonal matrix, and  $\sigma = 0$  for any singular matrix.
- Since  $V_n = \prod_{i=1}^n |r_{i,i}| = d(L)$  is a constant for a given  $L$ , we can improve  $\sigma$  by reducing  $\|b_i\|_2$ .
- Possible measurements other than the geometric mean?

Thank you!

Thank you!

Questions?