COMPUTER ENGINEERING

UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

# ADVANCED DIGITAL SIGNAL PROCESSING
# Chapter 7: MU-MIMO

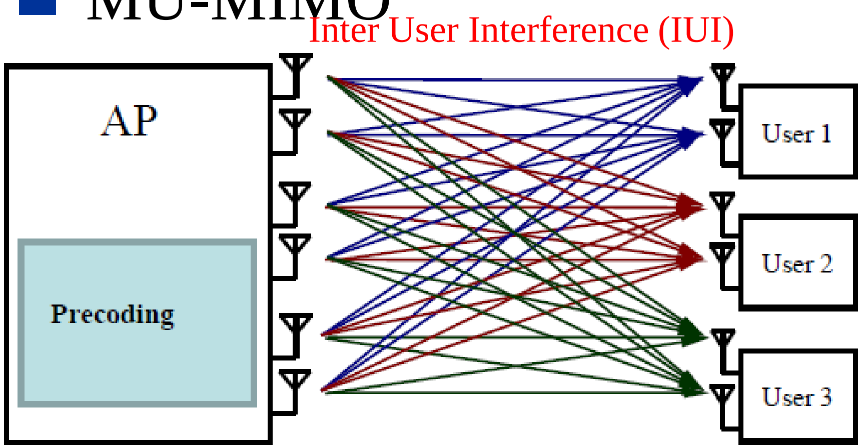
18/11/2017

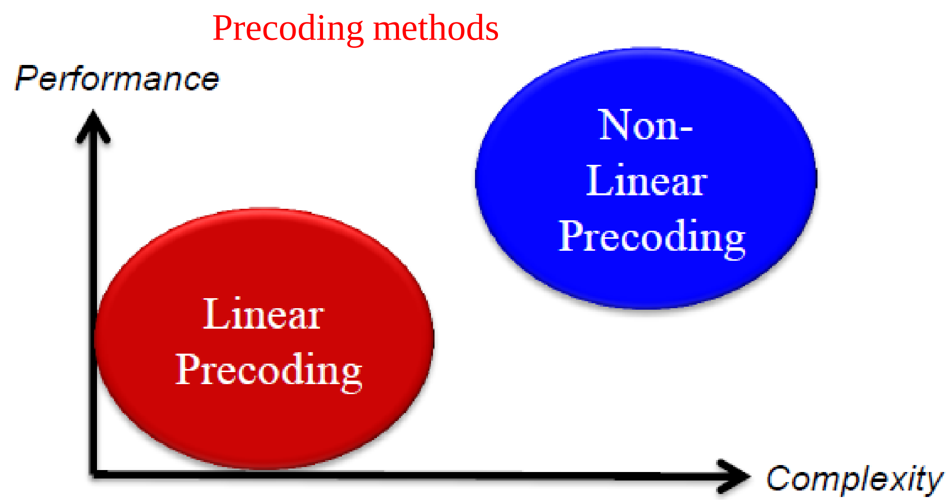
# Content

- Introduction
- Linear Precoding
  - Channel Inversion (CI)
  - Regularized Channel Inversion (RCI)
  - Block Diagonalization (BD)
- Non-Linear Precoding

# Introduction

- MU-MIMO

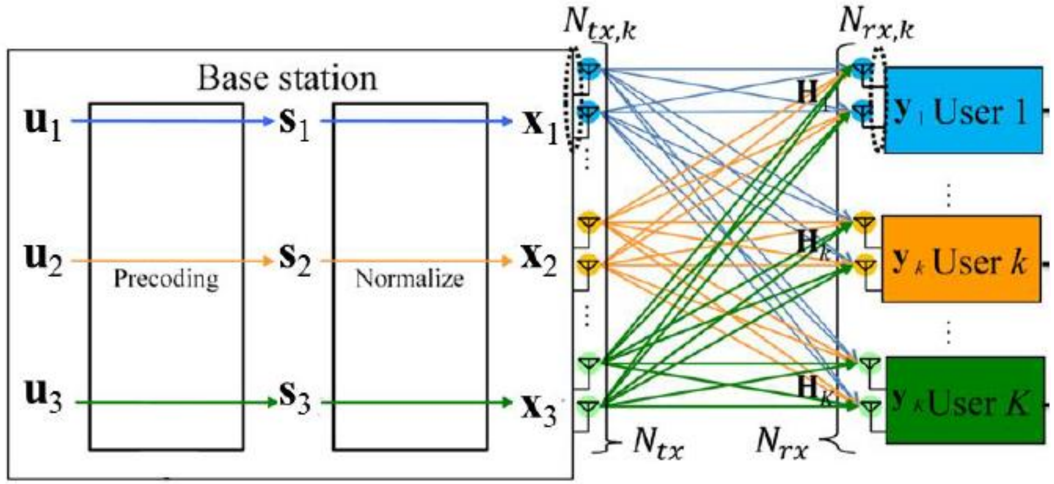



- There are two Precoding methods: Linear Precoding and Non-Linear Precoding.
  - Linear Precoding: is not complex but low performance.
    - Channel Inversion (CI)
    - Regularized Channel Inversion(RCI)
    - Block Diagonalization (BD)
  - Non-Linear Precoding: is complex but high performance.

# Introduction

- $N_{tx}$ : Number of transmission antenna
- $N_{tx,k}$ : Only User $k$
- $N_{rx}$ : Number of receive antenna
- $N_{rx,k}$ : Only User $k$
- *Number of stream is same $N_{rx,k}$
- $K$ : Number of User
- **u**: Transmission symbol vector ($N_{rx} \times 1$)
- **W** : Weight matrix ($N_{tx} \times N_{rx}$)
- **H** : Channel matrix ($N_{rx} \times N_{tx}$)
- $\mathbf{H}_k$ : Only User $k$ ($N_{rx,k} \times N_{tx}$)




Precoding

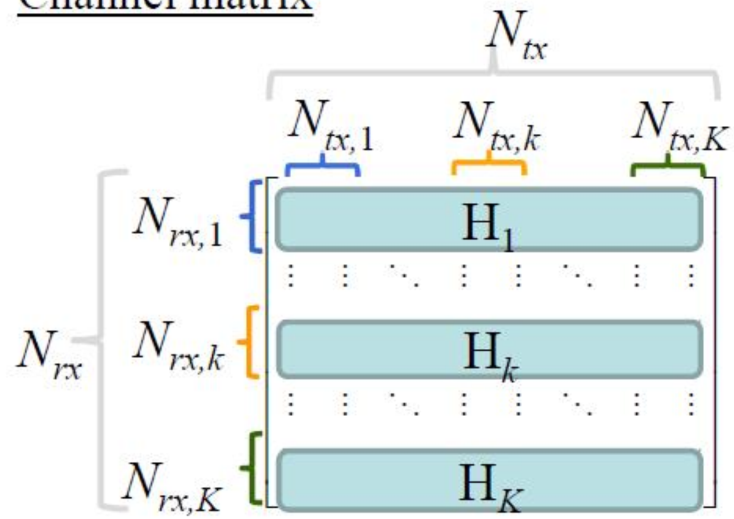
$$\mathbf{s} = \mathbf{W}\mathbf{u}$$

Normalize

$$\mathbf{x} = \frac{\mathbf{s}}{\sqrt{\gamma}}$$

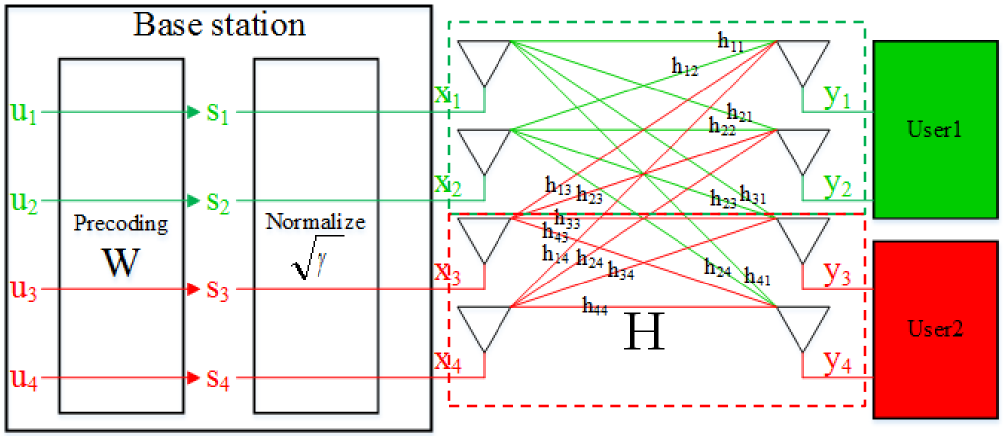
$$\gamma = E\left[\|\mathbf{s}\|^2\right]$$

Channel matrix



# Introduction

- MU-MIMO: 2 Users




Expected $H_1$ IUI $H_1$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}$$

Expected $H_2$ IUI $H_2$

- MU-MIMO Precoding: Remove IUI

$E[.]$: Expected value
$||A||$: Frobenius norm (Norm 2)
$||A|| = \sqrt{trace(AA^H)}$

# Channel Inversion (CI)

- CI use inverse matrix of **H** as weight matrix **W**.

$$W_{CI} = H^H(HH^H)^{-1}$$

- Precoding

$$\mathrm{s} = W_{CI}u = H^H(HH^H)^{-1}u$$

- Normalize

$$\mathrm{x} = \frac{s}{\sqrt{\gamma_{CI}}} = \frac{s}{\sqrt{\gamma_{CI}}} = \frac{H^H(HH^H)^{-1}u}{\sqrt{\gamma_{CI}}}$$

- Normalize gain

$$\gamma_{CI} = E[||s||^2] = E[||H^H(HH^H)^{-1}u||^2] \approx ||H^H(HH^H)^{-1}||^2$$

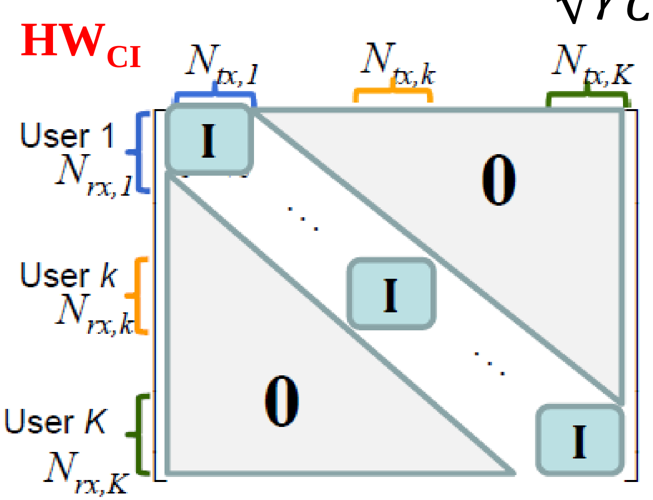
# Channel Inversion (CI)

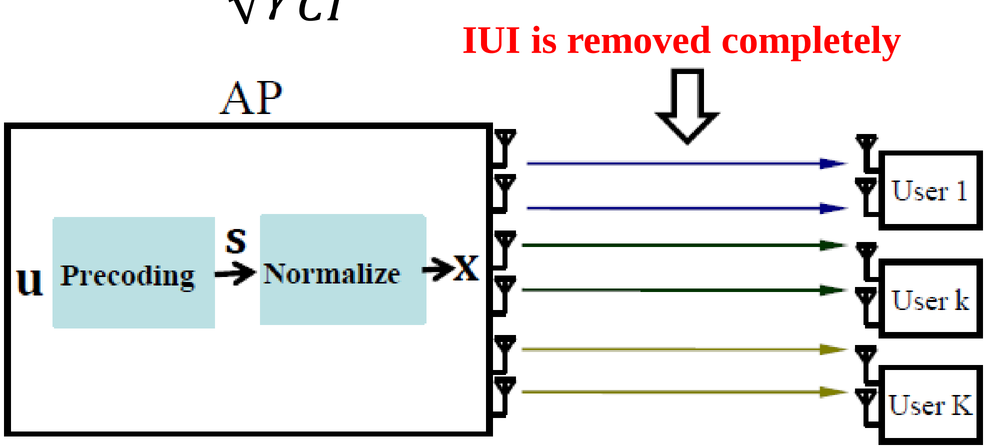
- Received signal

$$y = \frac{1}{\sqrt{\gamma_{CI}}} HW_{CI} u + n$$

$$= \frac{1}{\sqrt{\gamma_{CI}}} HH^H (HH^H)^{-1} u + n$$

$$= \frac{1}{\sqrt{\gamma_{CI}}} Iu + n = \frac{1}{\sqrt{\gamma_{CI}}} u + n$$



# Channel Inversion (CI)

- Decoding process

$$\sqrt{\gamma_{CI}}y = u + \sqrt{\gamma_{CI}}n$$

$$u = \sqrt{\gamma_{CI}}y \boxed{- \sqrt{\gamma_{CI}}n}$$

Noise

- Signal power of transmitted symbol $u$ is attenuated

# Regularized Channel Inversion(RCI)

- Regularized Channel Inversion weight matrix $\mathbf{W}_{RCI}$ is considering noise in addition to CI
- $\mathbf{W}_{RCI}$ is determined to **maximize SINR**.
- SINR (Signal to Interference plus Noise Ratio)

$$SINR = \frac{P}{I+N}$$

  - P: Received power
  - I: Interference power
  - N: Noise power

# Regularized Channel Inversion (RCI)

- RCI use inverse matrix of **H** as weight matrix **W**.

$$W_{RCI} = H^H(HH^H + \sigma^2 I)^{-1}$$

$\sigma^2$ → $E[nn^H]$: Noise power

- Precoding

$$\mathrm{s} = W_{RCI}u = H^H(HH^H + \sigma^2 I)^{-1}u$$

- Normalize

$$\mathrm{x} = \frac{s}{\sqrt{\gamma_{CI}}} = \frac{s}{\sqrt{\gamma_{CI}}} = \frac{H^H\left(HH^H+\sigma^2 I\right)^{-1}u}{\sqrt{\gamma_{CI}}}$$

- Normalize gain

$$\gamma_{RCI} = E[\|s\|^2] = E[\|H^H(HH^H + \sigma^2 I)^{-1}u\|^2]$$
$$\approx \|H^H(HH^H + \sigma^2 I)^{-1}\|^2$$

# Regularized Channel Inversion(RCI)

- Received signal $y = \frac{1}{\sqrt{\gamma_{RCI}}} HW_{RCI}u + n$

$$= \frac{1}{\sqrt{\gamma_{RCI}}} HH^H(HH^H + \sigma^2 I)^{-1}u + n$$

- Example: $N_{tx} \times N_{rx1}, N_{rx2}, N_{rx3}: 6 \times 2,2,2;\ \ \sigma^2 = 2$

**H**

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | -0.2828 + 1.9437i | -0.6718 + 2.7526i | 0.3616 - 1.1277i | 1.0393 + 1.2128i | -0.8305 - 0.2991i | -1.3233 + 0.9877i |
| 2 | 1.1522 - 1.0847i | 0.5756 + 0.1383i | -0.3519 + 0.0782i | 0.9109 + 0.4855i | -0.3523 - 0.8999i | 0.1283 + 0.3929i |
| 3 | -1.1465 + 0.2268i | -0.7781 - 1.9071i | 0.2695 + 2.1066i | -0.2397 + 1.0260i | -0.1748 + 0.6347i | -1.4424 + 0.1946i |
| 4 | 0.6737 + 1.0989i | -1.0636 - 0.3650i | -2.5644 - 0.7158i | 0.1810 + 0.8707i | -0.4807 + 0.0675i | 1.3025 + 0.2798i |
| 5 | -0.6691 + 0.1472i | 0.5530 - 0.8481i | 0.4659 - 0.2805i | 0.2442 - 0.3818i | 0.8368 - 0.1871i | 1.4099 + 0.0512i |
| 6 | -0.4003 + 2.2957i | -0.4234 - 0.7648i | 1.8536 + 1.1665i | 0.0964 + 0.4289i | 2.5383 + 0.2917i | -1.6625 - 0.7745i |

$\mathbf{HW_{RCI}}$

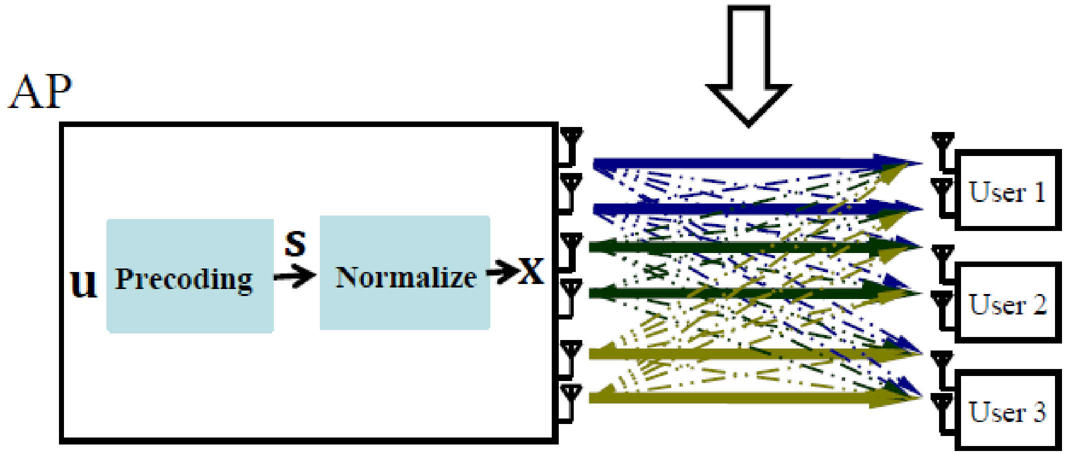
| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 0.8179 - 0.0000i | -0.0026 + 0.0429i | -0.0499 - 0.0902i | 0.0216 + 0.0436i | -0.1444 + 0.0304i | 0.0118 + 0.0118i |
| 2 | -0.0026 - 0.0429i | 0.4593 + 0.0000i | 0.0571 + 0.0829i | -0.0064 - 0.1333i | 0.0432 + 0.0845i | -0.1019 - 0.1852i |
| 3 | -0.0499 + 0.0902i | 0.0571 - 0.0829i | 0.7250 - 0.0000i | 0.0557 + 0.0309i | -0.1171 + 0.0983i | 0.1243 + 0.0581i |
| 4 | 0.0216 - 0.0436i | -0.0064 + 0.1333i | 0.0557 - 0.0309i | 0.7848 - 0.0000i | 0.0366 - 0.1279i | -0.1056 + 0.0375i |
| 5 | -0.1444 - 0.0304i | 0.0432 - 0.0845i | -0.1171 - 0.0983i | 0.0366 + 0.1279i | 0.4923 - 0.0000i | 0.0936 + 0.0734i |
| 6 | 0.0118 - 0.0118i | -0.1019 + 0.1852i | 0.1243 - 0.0581i | -0.1056 - 0.0375i | 0.0936 - 0.0734i | 0.7624 + 0.0000i |

# Regularized Channel Inversion(RCI)

$HW_{RCI}$

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 0.8179 - 0.0000i | -0.0026 + 0.0429i | -0.0499 - 0.0902i | 0.0216 + 0.0436i | -0.1444 + 0.0304i | 0.0118 + 0.0118i |
| 2 | -0.0026 - 0.0429i | 0.4593 + 0.0000i | 0.0571 + 0.0829i | -0.0064 - 0.1333i | 0.0432 + 0.0845i | -0.1019 - 0.1852i |
| 3 | -0.0499 + 0.0902i | 0.0571 - 0.0829i | 0.7250 - 0.0000i | 0.0557 + 0.0309i | -0.1171 + 0.0983i | 0.1243 + 0.0581i |
| 4 | 0.0216 - 0.0436i | -0.0064 + 0.1333i | 0.0557 - 0.0309i | 0.7848 - 0.0000i | 0.0366 - 0.1279i | -0.1056 + 0.0375i |
| 5 | -0.1444 - 0.0304i | 0.0432 - 0.0845i | -0.1171 - 0.0983i | 0.0366 + 0.1279i | 0.4923 - 0.0000i | 0.0936 + 0.0734i |
| 6 | 0.0118 - 0.0118i | -0.1019 + 0.1852 | 0.1243 - 0.0581i | -0.1056 - 0.0375i | 0.0936 - 0.0734i | 0.7624 + 0.0000i |

## IUI can not be removed completely



# Regularized Channel Inversion(RCI)

- Decoding process

$$\sqrt{\gamma_{RCI}}y = HW_{RCI}u + \sqrt{\gamma_{CI}}n$$

$$u \approx \sqrt{\gamma_{CI}}y - \sqrt{\gamma_{CI}}n$$

- Signal power of transmitted symbol *u* of RCI is attenuated, but less than CI since $\sqrt{\gamma_{RCI}} < \sqrt{\gamma_{CI}}$

MATLAB calculation result $(\sigma^2 = 2)$

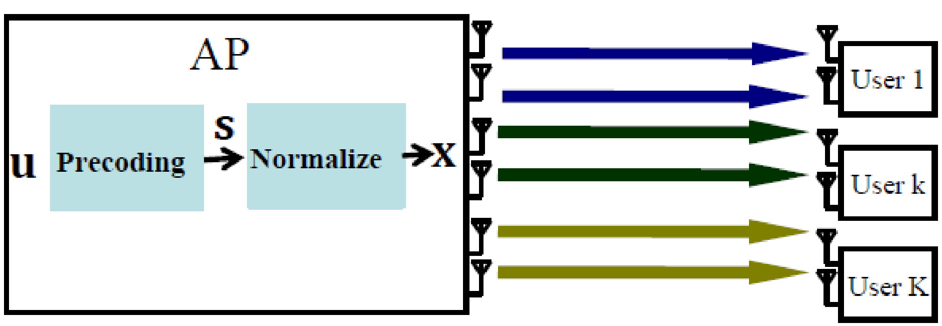
```
>> gamma_CI = norm(H' * inv(H * H'), 'fro')^2

gamma_CI =

  5.5113

>> gamma_RCI = norm(H' * inv(H * H' + sigma^2 * eye(6)),'fro')^2

gamma_RCI =

  0.3891
```

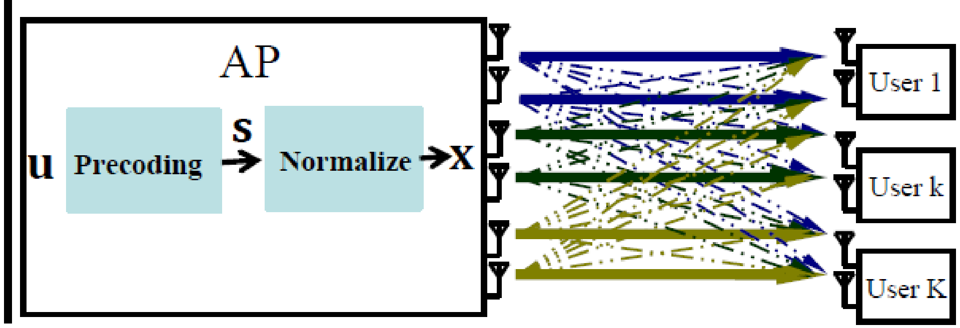
$$\gamma_{CI} \approx \|H^H(HH^H)^{-1}\|^2$$

$$\gamma_{RCI} \approx \|H^H(HH^H + \sigma^2 I)^{-1}\|^2$$

# Summary : CI & RCI

| | Channel Inversion | Regularized Channel Inversion |
|---|---|---|
| Weight matrix | $\mathbf{W}_{\mathrm{CI}} = \mathbf{H}^{\dagger} = \mathbf{H}^{H}\left(\mathbf{H}\mathbf{H}^{H}\right)^{-1}$ | $\mathbf{W}_{\mathrm{RCI}} = \mathbf{H}^{H}\left(\mathbf{H}\mathbf{H}^{H} + \sigma^{2}\mathbf{I}\right)^{-1}$ |
| Normalization gain | $\gamma_{\mathrm{CI}} \approx \left\Vert \mathbf{H}^{H}\left(\mathbf{H}\mathbf{H}^{H}\right)^{-1} \right\Vert^{2}$ | $\gamma_{\mathrm{RCI}} \approx \left\Vert \mathbf{H}^{H}\left(\mathbf{H}\mathbf{H}^{H} + \sigma^{2}\mathbf{I}\right)^{-1} \right\Vert^{2}$ |
| Noise enhancement | **Strong** | **weak** |
| IUI | **not occur** | **occur a little** |



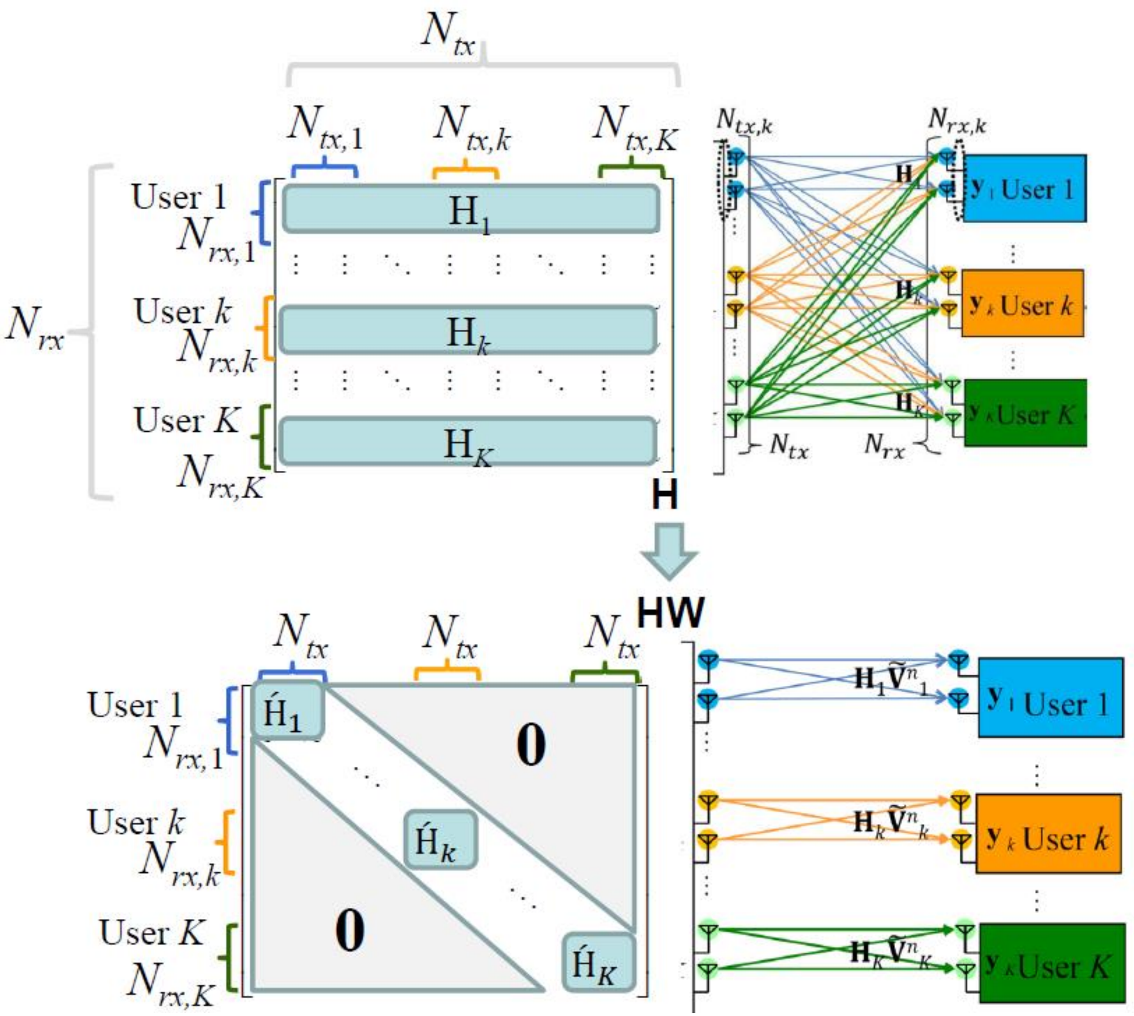
# Block Diagonalization (BD)

- Received symbol

$$y = \frac{1}{\sqrt{\gamma_{BD}}} H W_{BD} u + n$$

$$* \ \gamma_{BD} = E[\|W_{BD} u\|^2]$$

- H: Channel matrix
- $W_{BD}$: Weight matrix
- u: Transmitted symbol




- BD precoding: Remove IUI completely and low noise
- At receiver, MIMO-Decoding is performed

# Block Diagonalization (BD)

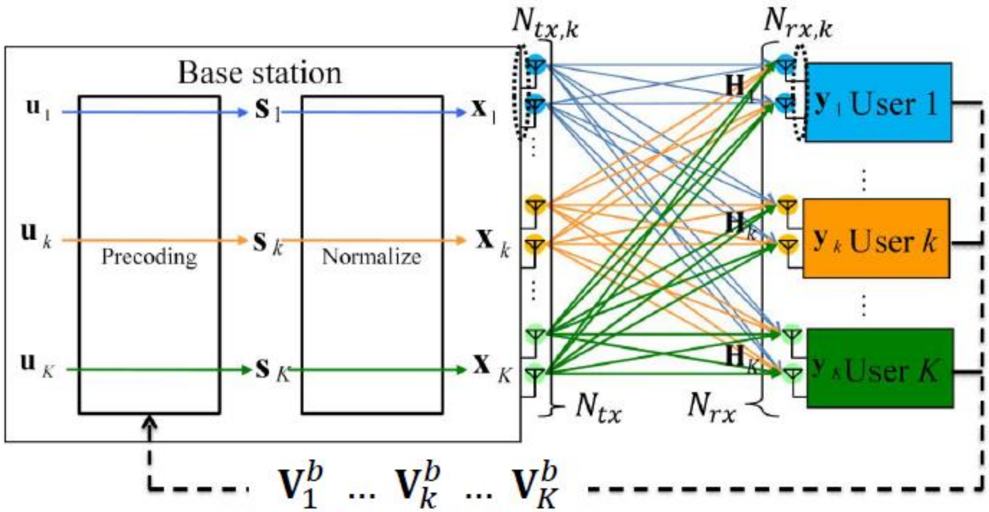
- SVD (Singular Value Decomposition)
  - $H_{m\times n} = U_{m\times m} \times \Sigma_{m\times n} \times V_{n\times n}^{H}$
  - V is orthogonal matrix, then $V^{-1} = V^{H}$
  - Then, $H_{m\times n} \times V_{n\times n} = U_{m\times m} \times \Sigma_{m\times n}$

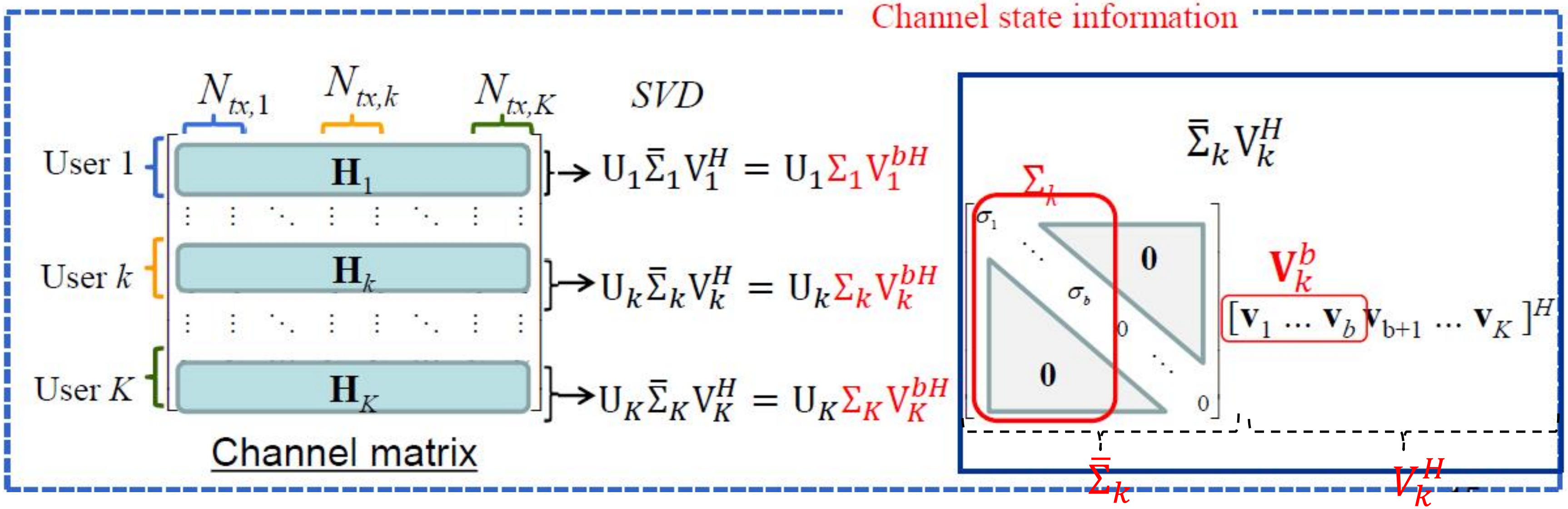
$$\underset{m\times n}{\mathbf{H}}\underset{n\times n}{\begin{bmatrix}\mathbf{v}_1 & \cdots & \mathbf{v}_n\end{bmatrix}} = \underset{m\times m}{\begin{bmatrix}\mathbf{u}_1 & \cdots & \mathbf{u}_m\end{bmatrix}} \underset{m\times n}{\begin{bmatrix}\sigma_1 & & & & \mathbf{0} \\ & \ddots & & & \\ & & \sigma_b & & \\ & & & 0 & \\ & & & & \ddots \\ \mathbf{0} & & & & & 0\end{bmatrix}} = \underset{m\times n}{\begin{bmatrix}\mathbf{u}_1\sigma_1 & \cdots & \mathbf{u}_b\sigma_b & \mathbf{0} & \cdots & \mathbf{0}\end{bmatrix}}$$

$$\mathbf{H}\begin{bmatrix}\mathbf{v}_1 & \cdots & \mathbf{v}_b & \underbrace{\mathbf{v}_{b+1} \quad \cdots \quad \mathbf{v}_n}_{\mathrm{NS}(\mathbf{H})}\end{bmatrix} = \begin{bmatrix}\mathbf{u}_1\sigma_1 & \cdots & \mathbf{u}_b\sigma_b & \mathbf{0} & \cdots & \mathbf{0}\end{bmatrix}$$

# Block Diagonalization (BD)

- Get $V_k^b$ as channel state information from each user



# Block Diagonalization (BD)

- $\widetilde{\mathbf{H}}_k$ is $\mathbf{H}_\text{v}$ (has no $\mathbf{V}_k^b$).

$$\mathbf{H}_\text{v} = [\,\mathbf{V}_1^b\ \mathbf{V}_2^b\ \ldots \cancel{\mathbf{V}_k^b}\ \ldots\ \mathbf{V}_K^b\,]^H$$

$$\downarrow$$

$$\widetilde{\mathbf{H}}_k = [\,\mathbf{V}_1^b\ \ldots \mathbf{V}_{k-1}^b \mathbf{V}_{k+1}^b\ \ldots\ \mathbf{V}_K^b\,]^H$$

$$\downarrow \text{ SVD}$$

$$\widetilde{\mathbf{H}}_k = \widetilde{\mathbf{U}}_k \widetilde{\mathbf{\Sigma}}_k \widetilde{\mathbf{V}}_k^H = \widetilde{\mathbf{U}}_k [\widetilde{\mathbf{\Sigma}}_k\ \mathbf{0}][\widetilde{\mathbf{V}}_k^b\ \widetilde{\mathbf{V}}_k^n]^H$$

Key point about BD method

- $\widetilde{\mathbf{V}}_k^n$ spans Null space of $\widetilde{\mathbf{H}}_k$ (※)

$$\widetilde{\mathbf{H}}_k \widetilde{\mathbf{V}}_k^n = [\mathbf{V}_1^{bH} \widetilde{\mathbf{V}}_k^n\ \ldots \mathbf{V}_{k-1}^{bH} \widetilde{\mathbf{V}}_k^n\ \ \mathbf{V}_{k+1}^{bH} \widetilde{\mathbf{V}}_k^n \cdots \mathbf{V}_K^{bH} \widetilde{\mathbf{V}}_k^n] = \mathbf{0}$$

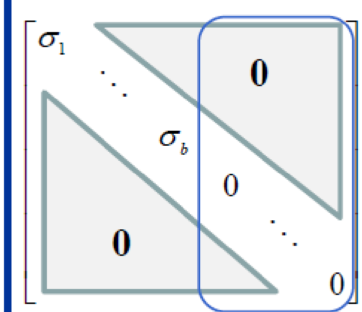
$$\downarrow$$

$$\boxed{\mathbf{V}_{k'}^{bH} \widetilde{\mathbf{V}}_k^n \begin{cases} \neq 0\ (k' = k) \\ = 0\ (k' \neq k) \end{cases} \quad \text{(a)}}$$

- Weight matrix $\mathbf{W}_\text{BD} = [\widetilde{\mathbf{V}}_1^n \widetilde{\mathbf{V}}_2^n \ldots \widetilde{\mathbf{V}}_K^n]$

※

$$\widetilde{\mathbf{H}}_k = \widetilde{\mathbf{U}}_k \widetilde{\mathbf{\Sigma}}_k \widetilde{\mathbf{V}}_k^H$$




$$[\mathbf{v}_1 \ldots \mathbf{v}_b\ \underbrace{\mathbf{v}_{b+1} \ldots \mathbf{v}_K}_{\widetilde{\mathbf{V}}_k^n}]^H$$

$\widetilde{\mathbf{V}}_k^n$ : NS($\widetilde{\mathbf{H}}_k$) $\rightarrow \widetilde{\mathbf{H}}_k \widetilde{\mathbf{V}}_k^n = \mathbf{0}$

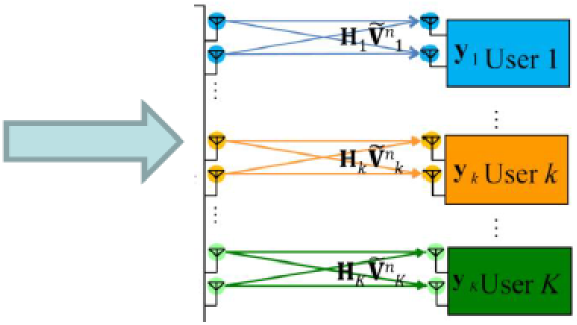
# Block Diagonalization (BD)

$$\mathbf{HW}_{BD} = \begin{bmatrix} \mathbf{H}_1\tilde{\mathbf{V}}_1^n & \mathbf{H}_1\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{H}_1\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{H}_1\tilde{\mathbf{V}}_K^n \\ \mathbf{H}_2\tilde{\mathbf{V}}_1^n & \mathbf{H}_2\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{H}_2\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{H}_2\tilde{\mathbf{V}}_K^n \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathbf{H}_k\tilde{\mathbf{V}}_1^n & \mathbf{H}_n\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{H}_k\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{H}_k\tilde{\mathbf{V}}_K^n \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathbf{H}_K\tilde{\mathbf{V}}_1^n & \mathbf{H}_K\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{H}_K\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{H}_K\tilde{\mathbf{V}}_K^n \end{bmatrix}$$

$$\boxed{\mathbf{V}_{k'}^{bH}\tilde{\mathbf{V}}_k^n \begin{cases} \neq 0 \ (k' = k) \\ = 0 \ (k' \neq k) \end{cases} \quad \text{(a)}}$$

$$= \begin{bmatrix} \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^{bH}\tilde{\mathbf{V}}_1^n & \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^{bH}\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^{bH}\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^{bH}\tilde{\mathbf{V}}_K^n \\ \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^{bH}\tilde{\mathbf{V}}_1^n & \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^{bH}\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^{bH}\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^{bH}\tilde{\mathbf{V}}_K^n \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^{bH}\tilde{\mathbf{V}}_1^n & \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^{bH}\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^{bH}\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^{bH}\tilde{\mathbf{V}}_K^n \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathbf{U}_K\boldsymbol{\Sigma}_K\mathbf{V}_K^{bH}\tilde{\mathbf{V}}_1^n & \mathbf{U}_K\boldsymbol{\Sigma}_K\mathbf{V}_K^{bH}\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{U}_K\boldsymbol{\Sigma}_K\mathbf{V}_K^{bH}\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{U}_K\boldsymbol{\Sigma}_K\mathbf{V}_K^{bH}\tilde{\mathbf{V}}_K^n \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{H}_1\tilde{\mathbf{V}}_1^n & \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{H}_2\tilde{\mathbf{V}}_2^n & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{H}_k\tilde{\mathbf{V}}_k^n & \cdots & \mathbf{O} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{H}_K\tilde{\mathbf{V}}_K^n \end{bmatrix}$$



# Block Diagonalization (BD)

- Example: 4x2,2

**H**(4x22)

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix} = \begin{bmatrix} \mathbf{H}1 \\ \mathbf{H}2 \end{bmatrix}$$

**H**1(rank:2)

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \end{bmatrix}$$

**H**2(rank:2)

$$\begin{bmatrix} h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix}$$

- Get channel state information($\mathbf{V}_1^b$, $\mathbf{V}_2^b$)
  - SVD (H1)

    $$\mathbf{U}_1\boldsymbol{\Sigma}_1[\underbrace{\boldsymbol{v}_{11} \quad \boldsymbol{v}_{12}}_{\mathbf{V}_1^b} \quad \boldsymbol{v}_{13} \quad \boldsymbol{v}_{14}]\ (\boldsymbol{v}_{11}\sim\boldsymbol{v}_{14} \in R^4)$$

  - SVD (H2)

    $$\mathbf{U}_2\boldsymbol{\Sigma}_2[\underbrace{\boldsymbol{v}_{21} \quad \boldsymbol{v}_{22}}_{\mathbf{V}_2^b} \quad \boldsymbol{v}_{23} \quad \boldsymbol{v}_{24}](\boldsymbol{v}_{21}\sim\boldsymbol{v}_{24} \in R^4)$$

# Block Diagonalization (BD)

- Example: 4x2,2

  - $\widetilde{\mathbf{H}}_1 = [\mathbf{V}_2^b]^H, \widetilde{\mathbf{H}}_2 = [\mathbf{V}_1^b]^H$ ($\widetilde{\mathbf{H}}_1, \widetilde{\mathbf{H}}_2$ : rank 2)
  - SVD ($\widetilde{\mathbf{H}}_1$)

$$\widetilde{\mathbf{U}}_1\widetilde{\mathbf{\Sigma}}_1 [\tilde{\boldsymbol{v}}_{11} \quad \tilde{\boldsymbol{v}}_{12} \quad \underbrace{\tilde{\boldsymbol{v}}_{13} \quad \tilde{\boldsymbol{v}}_{14}}_{\widetilde{\mathbf{V}}_1^n}] \; (\tilde{\boldsymbol{v}}_{11} \sim \tilde{\boldsymbol{v}}_{14} \in R^4)$$

$$\widetilde{\mathbf{H}}_1\widetilde{\mathbf{V}}_1^n = {\mathbf{V}_2^b}^H\widetilde{\mathbf{V}}_1^n = \mathbf{0}$$

  - SVD ($\widetilde{\mathbf{H}}_2$)

$$\widetilde{\mathbf{U}}_2\widetilde{\mathbf{\Sigma}}_2 [\tilde{\boldsymbol{v}}_{21} \quad \tilde{\boldsymbol{v}}_{22} \quad \underbrace{\tilde{\boldsymbol{v}}_{23} \quad \tilde{\boldsymbol{v}}_{24}}_{\widetilde{\mathbf{V}}_2^n}] \; (\tilde{\boldsymbol{v}}_{11} \sim \tilde{\boldsymbol{v}}_{14} \in R^4)$$

$$\widetilde{\mathbf{H}}_2\widetilde{\mathbf{V}}_2^n = {\mathbf{V}_1^b}^H\widetilde{\mathbf{V}}_2^n = \mathbf{0}$$

- Weight matrix **W**(4x4)

**W**=$[\widetilde{\mathbf{V}}_1^n \; \widetilde{\mathbf{V}}_2^n]$

- **HW** is

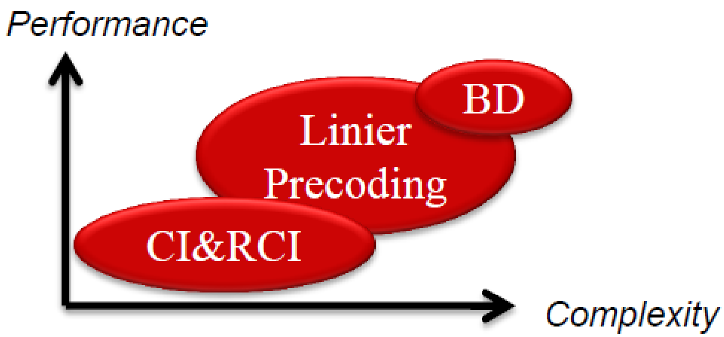
$$\begin{bmatrix}\mathbf{H1}\\ \mathbf{H2}\end{bmatrix}[\widetilde{\mathbf{V}}_1^n \; \widetilde{\mathbf{V}}_2^n]$$

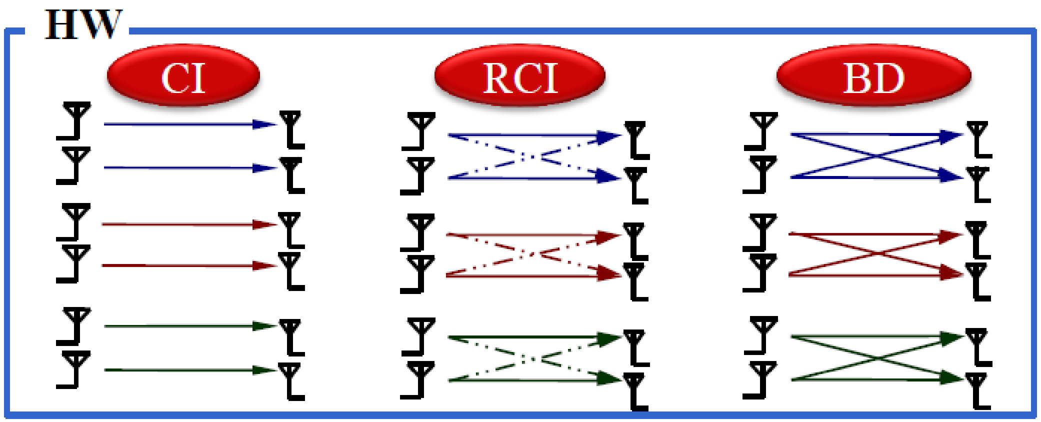
$$= \begin{bmatrix}\mathbf{U\Sigma V}_1^b\\ \mathbf{U\Sigma V}_2^b\end{bmatrix}[\widetilde{\mathbf{V}}_1^n \; \widetilde{\mathbf{V}}_2^n]$$

$$= \begin{bmatrix}\mathbf{U\Sigma V}_1^{bH}\widetilde{\mathbf{V}}_1^n & \mathbf{U\Sigma V}_1^{bH}\widetilde{\mathbf{V}}_2^n\\ \mathbf{U\Sigma V}_2^{bH}\widetilde{\mathbf{V}}_1^n & \mathbf{U\Sigma V}_2^{bH}\widetilde{\mathbf{V}}_2^n\end{bmatrix}$$

$$= \begin{bmatrix}\mathbf{U\Sigma V}_1^{bH}\widetilde{\mathbf{V}}_1^n & \mathbf{0}\\ \mathbf{0} & \mathbf{U\Sigma V}_2^{bH}\widetilde{\mathbf{V}}_2^n\end{bmatrix}$$

# Comparison CI , RCI - BD

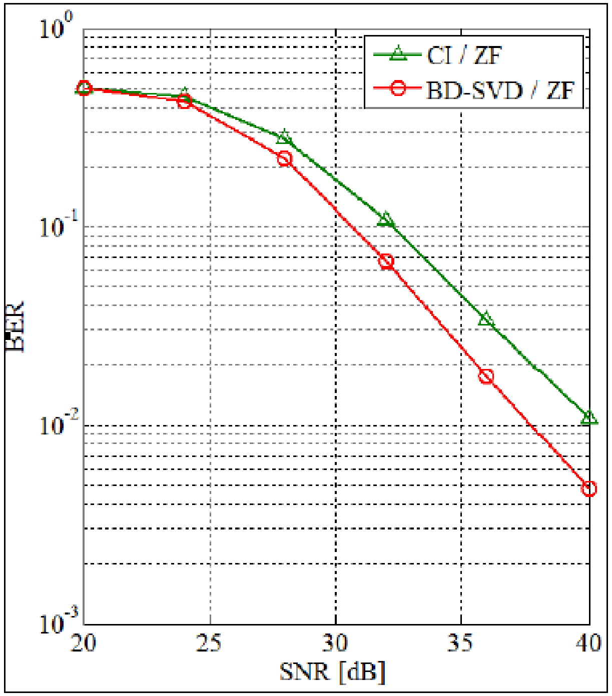


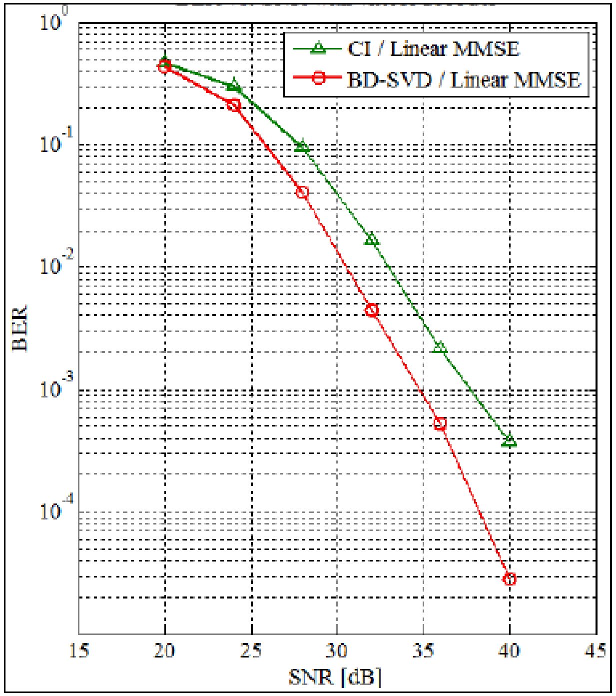




- BD is more complex than CI , Because BD needs *SVD*.
- CI has no Diversity Gain like above picture.
- RCI has Diversity Gain , but it's smaller than BD.

# BER Characteristic (CI - BD)

- BER Characteristic: 4x2,2




- The BER of BD is always better than that of CI

# Summary

- MU-MIMO Precoding
  - Channel Inversion
  - Regularized Channel Inversion
  - Block Diagonalization

# END