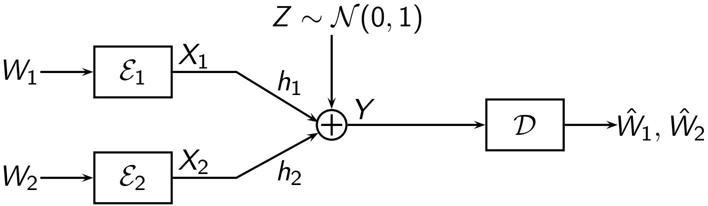
On Compute-and-Forward with Feedback

Or Ordentlich
Joint work with Uri Erez and Bobak Nazer

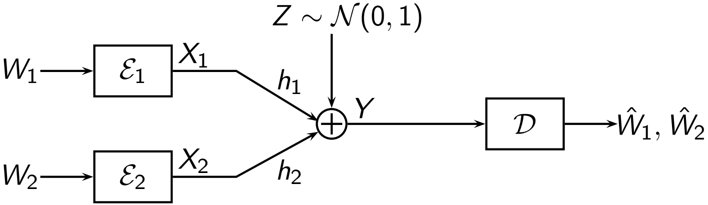
Information Theory Workshop
Jerusalem, Israel
April 27, 2015

Gaussian Multiple-Access Channel



$$Y = h_1 X_1 + h_2 X_2 + Z$$

Gaussian Multiple-Access Channel



$$Y = h_1 X_1 + h_2 X_2 + Z$$

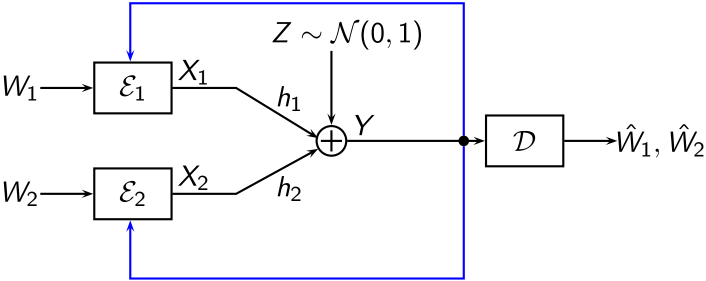
Capacity Region

$$R_1 < \frac{1}{2}\log(1 + h_1^2 \mathsf{SNR})$$

$$R_2 < \frac{1}{2}\log(1 + h_2^2 \mathsf{SNR})$$

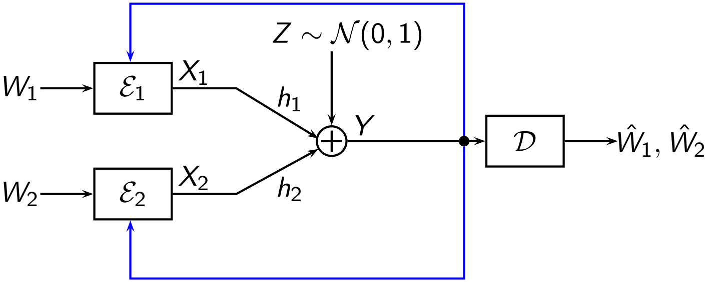
$$R_1 + R_2 < \frac{1}{2}\log(1 + \|\mathbf{h}\|^2 \mathsf{SNR})$$

Gaussian Multiple-Access Channel



$$Y = h_1 X_1 + h_2 X_2 + Z$$

Gaussian Multiple-Access Channel



$$Y = h_1 X_1 + h_2 X_2 + Z$$

Feedback Capacity Region (Ozarow 84)

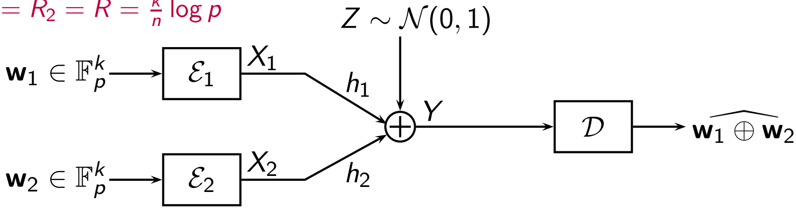
$$R_1 < \frac{1}{2}\log(1 + (1-\rho^2)h_1^2 \mathsf{SNR})$$

$$R_2 < \frac{1}{2}\log(1 + (1-\rho^2)h_2^2 \mathsf{SNR})$$

$$R_1 + R_2 < \frac{1}{2}\log(1 + (\|\mathbf{h}\|^2 + 2\rho|h_1 h_2|)\mathsf{SNR})$$

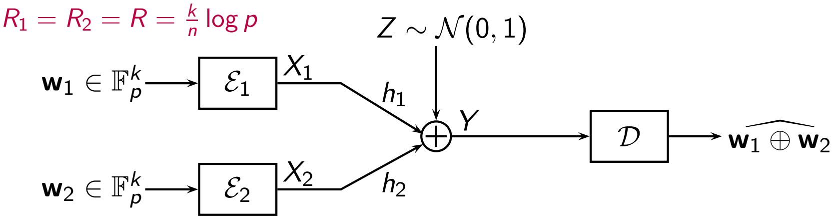
Gaussian MAC - Compute-and-Forward

$R_1 = R_2 = R = \frac{k}{n} \log p$



$$Y = h_1 X_1 + h_2 X_2 + Z$$

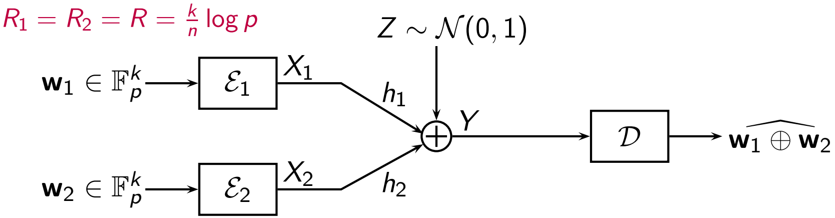
Gaussian MAC - Compute-and-Forward



$$Y = h_1 X_1 + h_2 X_2 + Z$$

- We assume full CSI everywhere

Gaussian MAC - Compute-and-Forward



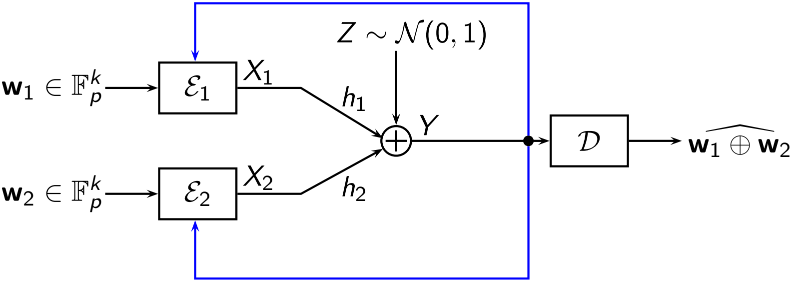
$$Y = h_1X_1 + h_2X_2 + Z$$

- We assume full CSI everywhere
- Only lower and upper bounds are known (Nazer & Gastpar 11)

$$\frac{1}{2}\log\left(\frac{1}{2} + \min\{h_1^2, h_2^2\}\mathrm{SNR}\right) \leq C_{\mathrm{comp}} \leq \frac{1}{2}\log\left(1 + \min\{h_1^2, h_2^2\}\mathrm{SNR}\right)$$

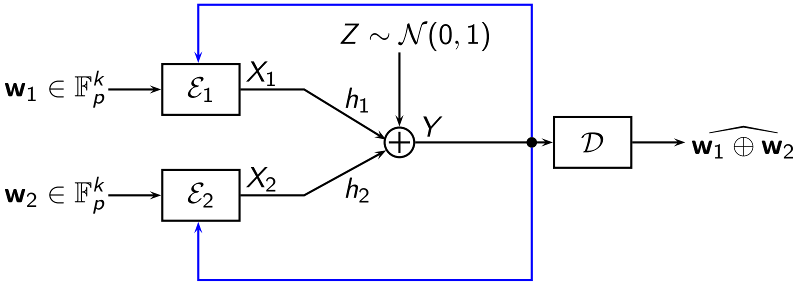
- At high SNR the bounds coincide. At low SNR separation is optimal

Gaussian MAC - Compute-and-Forward



$$Y = h_1 X_1 + h_2 X_2 + Z$$

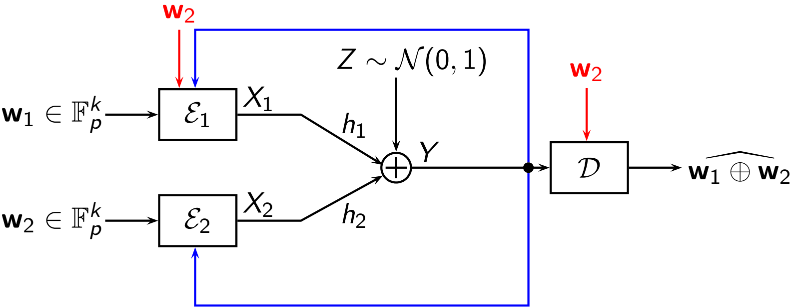
Gaussian MAC - Compute-and-Forward



$$Y = h_1 X_1 + h_2 X_2 + Z$$

How Much Does Feedback Help?

Gaussian MAC - Compute-and-Forward

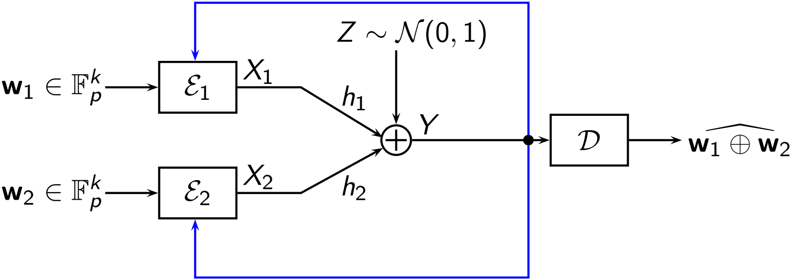


$$Y = h_1X_1 + h_2X_2 + Z$$

How Much Does Feedback Help?

- Upper bound remains the same $C_{\text{comp}} \leq \frac{1}{2}\log\left(1 + \min\{h_1^2, h_2^2\}\text{SNR}\right)$

Gaussian MAC - Compute-and-Forward

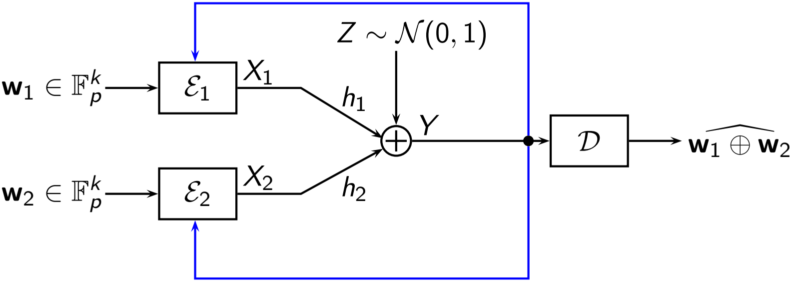


$$Y = h_1X_1 + h_2X_2 + Z$$

How Much Does Feedback Help?

- Upper bound remains the same $C_{\text{comp}} \leq \frac{1}{2}\log\left(1 + \min\{h_1^2, h_2^2\}\text{SNR}\right)$
- No non-trivial lower bounds are known

Gaussian MAC - Compute-and-Forward



$$Y = h_1X_1 + h_2X_2 + Z$$

How Much Does Feedback Help?

- Upper bound remains the same $C_{\text{comp}} \leq \frac{1}{2}\log\left(1 + \min\{h_1^2, h_2^2\}\text{SNR}\right)$
- No non-trivial lower bounds are known
- In this work we derive a novel lower bound

Gaussian MAC - Compute-and-Forward

Main Result

For any $0 < \rho \leq 1$ let

$$\rho_1 = 1 - (1-\rho)\left(\frac{h_2}{h_1}\right)^2,$$

$$R_c = \frac{1}{2}\log^+\left(\frac{1}{2} + (1-\rho)h_2^2\mathsf{SNR}\right),$$

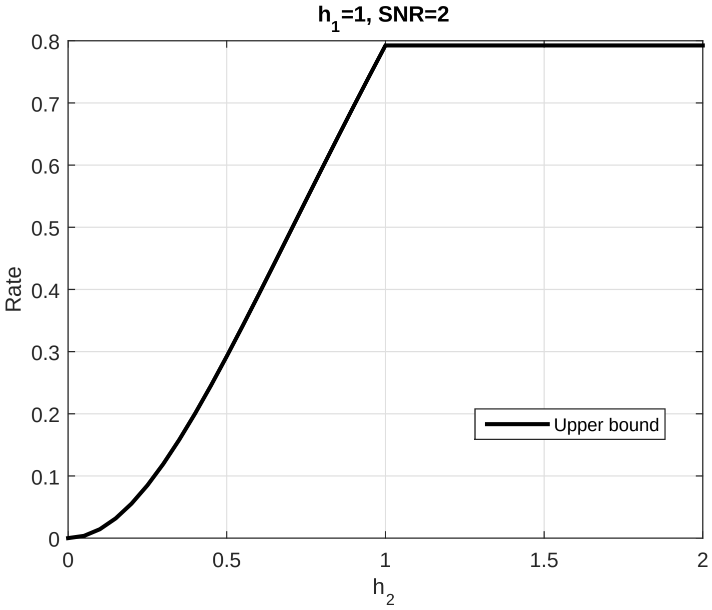
$$R' = \frac{1}{2}\log\left(1 + \frac{(h_1\sqrt{\rho_1} + h_2\sqrt{\rho})^2\mathsf{SNR}}{1 + 2(1-\rho)h_2^2\mathsf{SNR}}\right).$$

Any computation rate satisfying

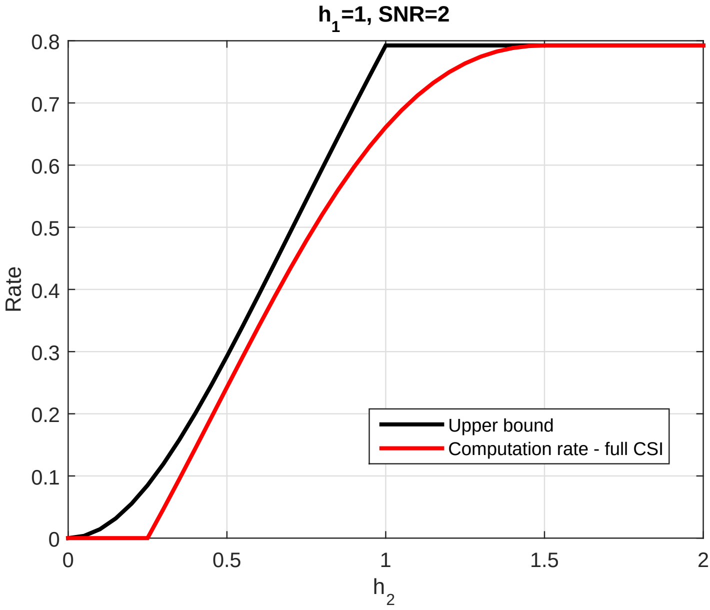
$$R < \max_{0<\rho\leq 1} \min\left(R' + R_c, \frac{1}{2}\log\left(1 + (1-\rho)h_2^2\mathsf{SNR}\right)\right)$$

is achievable with feedback.

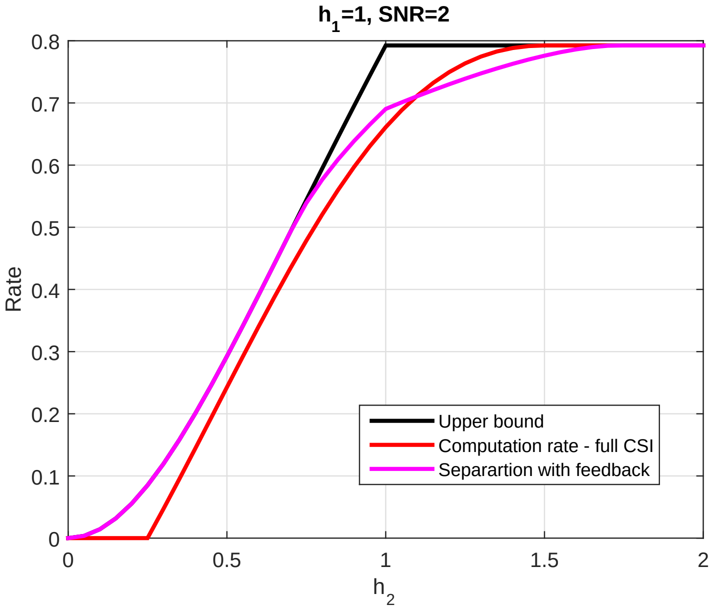
Gaussian MAC - Compute-and-Forward



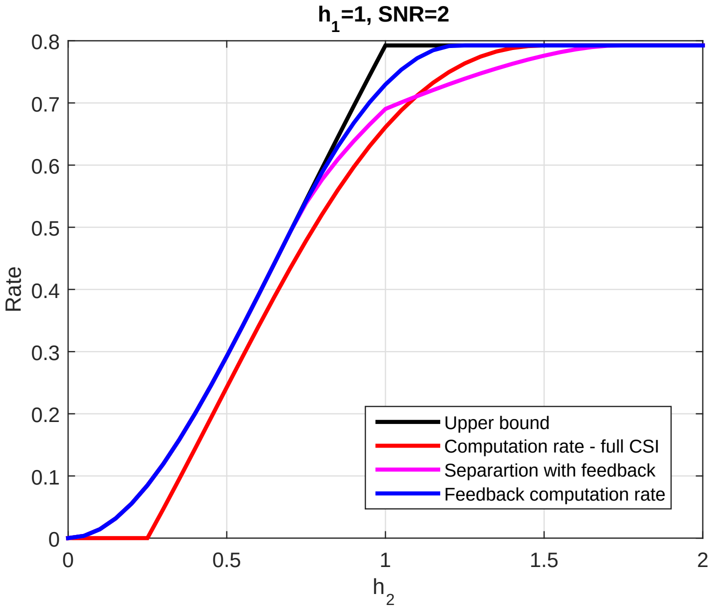
Gaussian MAC - Compute-and-Forward



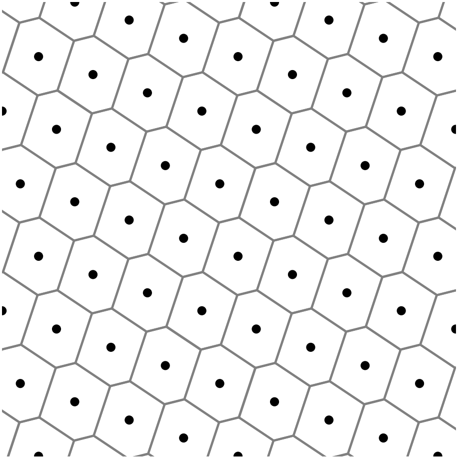
Gaussian MAC - Compute-and-Forward



Gaussian MAC - Compute-and-Forward

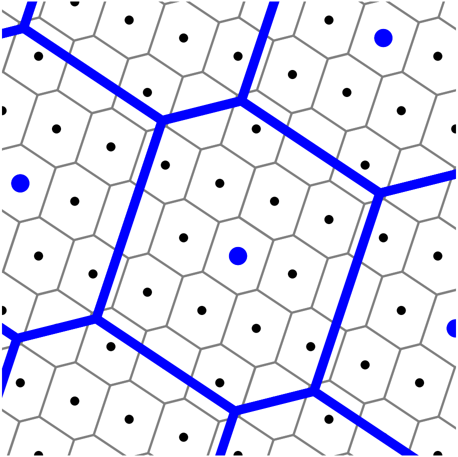


Lattice List Decoder (Song & Devroye 13)



Fine lattice Λ_c

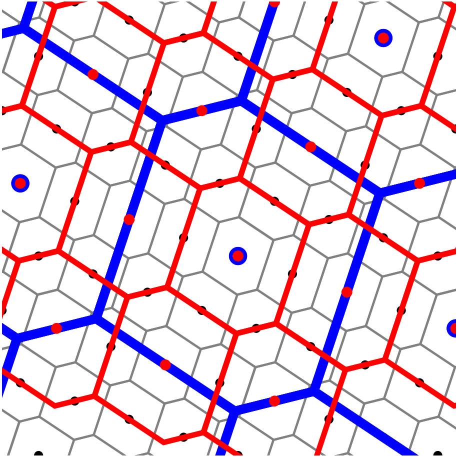
Lattice List Decoder (Song & Devroye 13)



Fine lattice Λ_c, coarse lattice $\Lambda \subseteq \Lambda_c$

$$\mathcal{C} = \Lambda_c \cap \mathcal{V}$$

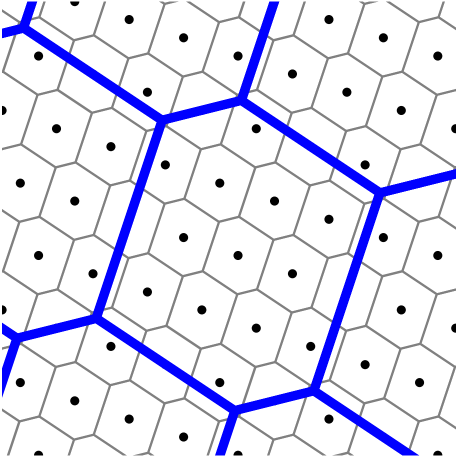
Lattice List Decoder (Song & Devroye 13)



Fine lattice Λ_c, coarse lattice Λ, intermediate lattice Λ_s, $\Lambda \subseteq \Lambda_s \subseteq \Lambda_c$

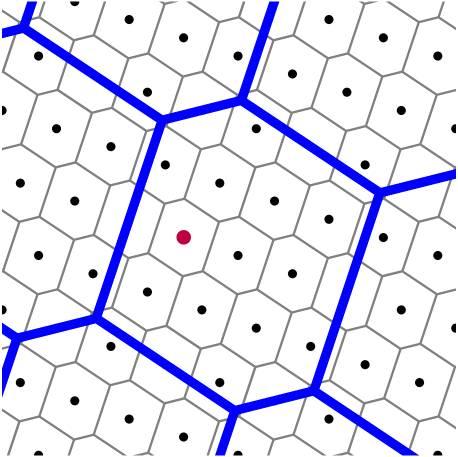
$$\mathcal{C} = \Lambda_c \cap \mathcal{V}$$

Lattice List Decoder (Song & Devroye 13)



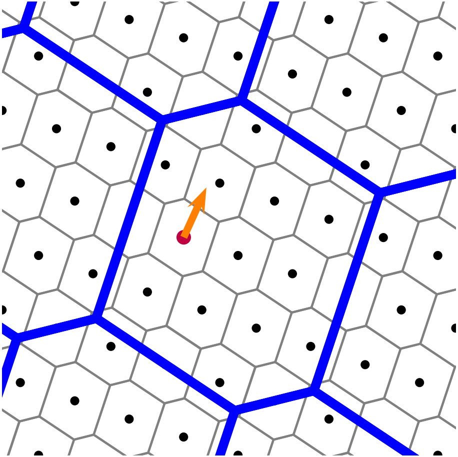
AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$

Lattice List Decoder (Song & Devroye 13)

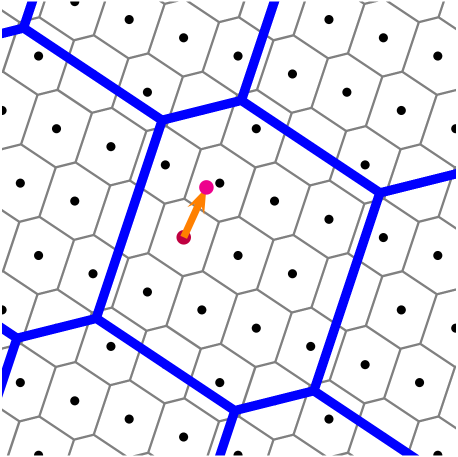


AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$

Lattice List Decoder (Song & Devroye 13)

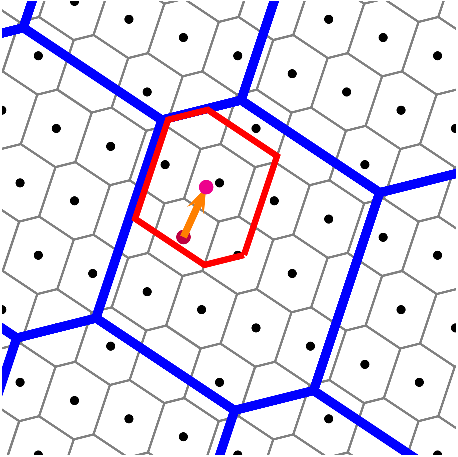


AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$



AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$

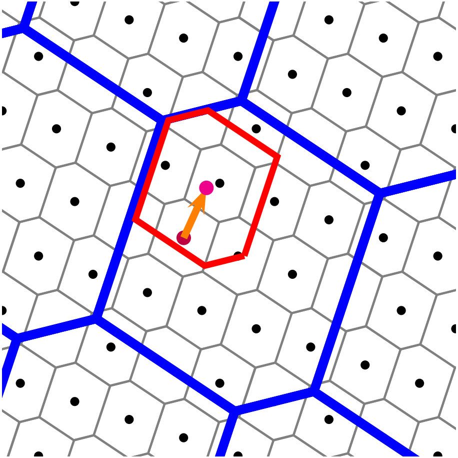
Lattice List Decoder (Song & Devroye 13)



AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$

Decode a list of codewords: $L = \{\mathbf{c} \in \mathcal{C} \; : \mathbf{c} \in [\mathbf{y} + \mathcal{V}_s] \bmod \Lambda\}$

Lattice List Decoder (Song & Devroye 13)

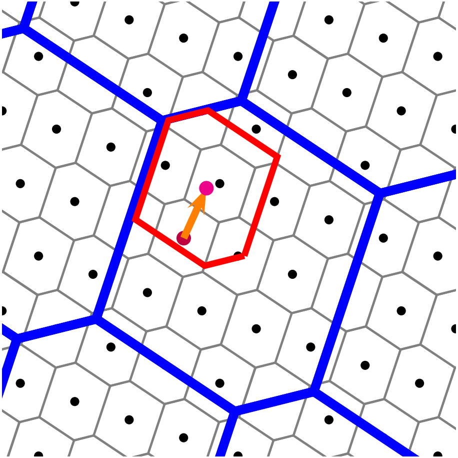


AWGN channel $\mathbf{y} = \mathbf{x} + \mathbf{z}$, $\quad R > C$

Decode a list of codewords: $L = \{\mathbf{c} \in \mathcal{C} \ : \mathbf{c} \in [\mathbf{y} + \mathcal{V}_s] \bmod \Lambda\}$

$|L| = \log\left(\frac{\mathrm{Vol}(\mathcal{V}_s)}{\mathrm{Vol}(\mathcal{V}_c)}\right)$

Lattice List Decoder (Song & Devroye 13)



Theorem (Song & Devroye 13)

It is possible to decode a list with size $2^{n(R-C)}$ that contains the true codeword w.h.p. using a lattice list decoder

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

- Block Markov coding

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user i can decode $\mathbf{w}_{\bar{i}}$ using the feedback link

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user i can decode $\mathbf{w}_{\bar{i}}$ using the feedback link
- In each block, each user superimposes encoding of a new message and encoding of the sum of messages from the last block
- The encoding of the sum is transmitted **coherently**

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user i can decode $\mathbf{w}_{\bar{i}}$ using the feedback link
- In each block, each user superimposes encoding of a new message and encoding of the sum of messages from the last block
- The encoding of the sum is transmitted **coherently**
- The receiver decodes the coherent part first, and then a list of candidates for the new sum

Compute-and-Forward with Feedback - High-Level

High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user i can decode $\mathbf{w}_{\bar{i}}$ using the feedback link
- In each block, each user superimposes encoding of a new message and encoding of the sum of messages from the last block
- The encoding of the sum is transmitted **coherently**
- The receiver decodes the coherent part first, and then a list of candidates for the new sum

A compute-and-forward variant of Cover-Leung 81

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)},\ k = 1, \dots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $\mathcal{C}$, such that $\tilde{\mathbf{x}}_i^{(k)} = f\left(\mathbf{w}_i^{(k)}\right) \in \mathcal{C}$

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)},\ k = 1, \ldots, N$ over $N+1$ blocks
- Both users encode their messages using the same lattice code $\mathcal{C}$, such that $\tilde{\mathbf{x}}_i^{(k)} = f\left(\mathbf{w}_i^{(k)}\right) \in \mathcal{C}$
- First block: $\mathbf{x}_i^{(1)} = \sqrt{1-\rho}\tilde{\mathbf{x}}_i^{(1)}$

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $\mathcal{C}$, such that $\tilde{\mathbf{x}}_i^{(k)} = f\left(\mathbf{w}_i^{(k)}\right) \in \mathcal{C}$
- First block: $\mathbf{x}_i^{(1)} = \sqrt{1-\rho}\tilde{\mathbf{x}}_i^{(1)}$
- Receiver sees

$$\mathbf{y}^{(1)} = \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(1)} + \tilde{\mathbf{x}}_2^{(1)}\right) + \mathbf{z}^{(1)}.$$

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)}, \ k = 1, \ldots, N$ over $N+1$ blocks
- Both users encode their messages using the same lattice code $\mathcal{C}$, such that $\tilde{\mathbf{x}}_i^{(k)} = f\left(\mathbf{w}_i^{(k)}\right) \in \mathcal{C}$
- First block: $\mathbf{x}_i^{(1)} = \sqrt{1-\rho}\tilde{\mathbf{x}}_i^{(1)}$
- Receiver sees

$$\mathbf{y}^{(1)} = \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(1)} + \tilde{\mathbf{x}}_2^{(1)}\right) + \mathbf{z}^{(1)}.$$

- $R < R_{\text{comp}} \triangleq \frac{1}{2}\log\left(\frac{1}{2} + (1-\rho)\text{SNR}\right)$ is needed for decoding $\mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$

Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $\mathbf{w}_1^{(k)} \oplus \mathbf{w}_2^{(k)}, \; k = 1, \dots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $\mathcal{C}$, such that $\tilde{\mathbf{x}}_i^{(k)} = f\left(\mathbf{w}_i^{(k)}\right) \in \mathcal{C}$
- First block: $\mathbf{x}_i^{(1)} = \sqrt{1-\rho}\tilde{\mathbf{x}}_i^{(1)}$
- Receiver sees

$$\mathbf{y}^{(1)} = \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(1)} + \tilde{\mathbf{x}}_2^{(1)}\right) + \mathbf{z}^{(1)}.$$

- $R < R_{\text{comp}} \triangleq \frac{1}{2}\log\left(\frac{1}{2} + (1-\rho)\text{SNR}\right)$ is needed for decoding $\mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$
- In our case $R > R_{\text{comp}}$ and the receiver can decode a list $L^{(1)}$ of candidates for $\mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$ with size $|L^{(1)}| = 2^{n(R-R_{\text{comp}})}$

Compute-and-Forward with Feedback

- Using the feedback link, user i can decode $\mathbf{w}_{\bar{i}}^{(1)}$ if

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

Compute-and-Forward with Feedback

- Using the feedback link, user i can decode $\mathbf{w}_{\bar{i}}^{(1)}$ if

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathrm{SNR})$$

- Both users can compute $\mathbf{v}^{(1)} = \mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$

Compute-and-Forward with Feedback

- Using the feedback link, user i can decode $\mathbf{w}_{\bar{i}}^{(1)}$ if

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

- Both users can compute $\mathbf{v}^{(1)} = \mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$
- Both users apply the same binning function $\mathbf{B} : [2^{nR}] \mapsto [2^{nR'}]$, $R' < R$, to obtain $\mathbf{B}\left(\mathbf{v}^{(1)}\right)$

Compute-and-Forward with Feedback

- Using the feedback link, user i can decode $\mathbf{w}_i^{(1)}$ if

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

- Both users can compute $\mathbf{v}^{(1)} = \mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$
- Both users apply the same binning function $\mathbf{B} : [2^{nR}] \mapsto [2^{nR'}]$, $R' < R$, to obtain $\mathbf{B}\left(\mathbf{v}^{(1)}\right)$
- Each user encodes $\mathbf{B}\left(\mathbf{v}^{(1)}\right)$ to $\mathbf{x}_{\mathsf{cohr}}^{(1)}$ using the same codebook $\mathcal{C}'$ with rate R' and average power SNR

Compute-and-Forward with Feedback

- Using the feedback link, user i can decode $\mathbf{w}_i^{(1)}$ if

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

- Both users can compute $\mathbf{v}^{(1)} = \mathbf{w}_1^{(1)} \oplus \mathbf{w}_2^{(1)}$
- Both users apply the same binning function $\mathbf{B} : [2^{nR}] \mapsto [2^{nR'}]$, $R' < R$, to obtain $\mathbf{B}\left(\mathbf{v}^{(1)}\right)$
- Each user encodes $\mathbf{B}\left(\mathbf{v}^{(1)}\right)$ to $\mathbf{x}_{\mathsf{cohr}}^{(1)}$ using the same codebook $\mathcal{C}'$ with rate R' and average power SNR
- In addition, each user encodes a new message $\mathbf{w}_i^{(2)}$ to the codeword $\tilde{\mathbf{x}}_i^{(2)}$ and transmits

$$\mathbf{x}_i^{(2)} = \sqrt{\rho}\mathbf{x}_{\mathsf{cohr}}^{(1)} + \sqrt{1-\rho}\tilde{\mathbf{x}}_i^{(2)}$$

Compute-and-Forward with Feedback

- Channel output is

$$\mathbf{y}^{(2)} = 2\sqrt{\rho}\mathbf{x}_{\text{cohr}}^{(1)} + \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(2)} + \tilde{\mathbf{x}}_2^{(2)}\right) + \mathbf{z}^{(2)}$$

Compute-and-Forward with Feedback

- Channel output is

$$\mathbf{y}^{(2)} = 2\sqrt{\rho}\mathbf{x}_{\text{cohr}}^{(1)} + \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(2)} + \tilde{\mathbf{x}}_2^{(2)}\right) + \mathbf{z}^{(2)}$$

- Can decode $\mathbf{x}_{\text{cohr}}^{(1)}$ if

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\text{SNR}}{1 + 2(1-\rho)\text{SNR}}\right)$$

Compute-and-Forward with Feedback

- Channel output is

$$\mathbf{y}^{(2)} = 2\sqrt{\rho}\mathbf{x}_{\text{cohr}}^{(1)} + \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(2)} + \tilde{\mathbf{x}}_2^{(2)}\right) + \mathbf{z}^{(2)}$$

- Can decode $\mathbf{x}_{\text{cohr}}^{(1)}$ if

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\text{SNR}}{1 + 2(1-\rho)\text{SNR}}\right)$$

- The decoder looks for a unique $\mathbf{w} \in \mathbb{F}_p^k$ in $L^{(1)} \cap \mathbf{B}^{-1}\left(\mathbf{v}^{(1)}\right)$

Compute-and-Forward with Feedback

- Channel output is

$$\mathbf{y}^{(2)} = 2\sqrt{\rho}\mathbf{x}_{\text{cohr}}^{(1)} + \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(2)} + \tilde{\mathbf{x}}_2^{(2)}\right) + \mathbf{z}^{(2)}$$

- Can decode $\mathbf{x}_{\text{cohr}}^{(1)}$ if

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\text{SNR}}{1+2(1-\rho)\text{SNR}}\right)$$

- The decoder looks for a unique $\mathbf{w} \in \mathbb{F}_p^k$ in $L^{(1)} \cap \mathbf{B}^{-1}\left(\mathbf{v}^{(1)}\right)$
- If $R' > R - R_{\text{comp}}$ such a $\mathbf{w} \in \mathbb{F}_p^k$ will be found with probability 1

Compute-and-Forward with Feedback

- Channel output is

$$\mathbf{y}^{(2)} = 2\sqrt{\rho}\mathbf{x}_{\text{cohr}}^{(1)} + \sqrt{1-\rho}\left(\tilde{\mathbf{x}}_1^{(2)} + \tilde{\mathbf{x}}_2^{(2)}\right) + \mathbf{z}^{(2)}$$

- Can decode $\mathbf{x}_{\text{cohr}}^{(1)}$ if

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\text{SNR}}{1+2(1-\rho)\text{SNR}}\right)$$

- The decoder looks for a unique $\mathbf{w} \in \mathbb{F}_p^k$ in $L^{(1)} \cap \mathbf{B}^{-1}\left(\mathbf{v}^{(1)}\right)$
- If $R' > R - R_{\text{comp}}$ such a $\mathbf{w} \in \mathbb{F}_p^k$ will be found with probability 1
- Next, the decoder subtracts $\mathbf{x}_{\text{cohr}}^{(1)}$ from $\mathbf{y}^{(2)}$ and decodes a list $L^{(2)}$ of candidates for $\mathbf{v}^{(2)} = \mathbf{w}_1^{(2)} \oplus \mathbf{w}_2^{(2)}$

Compute-and-Forward with Feedback - Rate Constraints

- Correct decoding through feedback link

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

Compute-and-Forward with Feedback - Rate Constraints

- Correct decoding through feedback link

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

- Correct decoding of $\mathbf{x}_{\mathsf{cohr}}$

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\mathsf{SNR}}{1 + 2(1-\rho)\mathsf{SNR}}\right)$$

Compute-and-Forward with Feedback - Rate Constraints

- Correct decoding through feedback link

$$R < \frac{1}{2}\log(1 + (1-\rho)\mathsf{SNR})$$

- Correct decoding of $\mathbf{x}_{\mathsf{cohr}}$

$$R' \leq \frac{1}{2}\log\left(1 + \frac{4\rho\mathsf{SNR}}{1 + 2(1-\rho)\mathsf{SNR}}\right)$$

- Unique element in intersection of list and bin

$$R' > R - \frac{1}{2}\log\left(\frac{1}{2} + (1-\rho)\mathsf{SNR}\right)$$

Compute-and-Forward with Feedback - Rate Constraints

Achievable Rate

$$R < \min\left\{\frac{1}{2}\log(1+(1-\rho)\mathsf{SNR}),\right.$$
$$\left.\frac{1}{2}\log\left(1+\frac{4\rho\mathsf{SNR}}{1+2(1-\rho)\mathsf{SNR}}\right)+\frac{1}{2}\log\left(\frac{1}{2}+(1-\rho)\mathsf{SNR}\right)\right\}$$

Compute-and-Forward with Feedback - Rate Constraints

Achievable Rate

$$R < \min\left\{\frac{1}{2}\log(1+(1-\rho)\mathrm{SNR}),\right.$$
$$\left.\frac{1}{2}\log\left(\frac{1}{2}+(1+\rho)\mathrm{SNR}\right)\right\}$$

Compute-and-Forward with Feedback - Rate Constraints

Achievable Rate

$$R < \min\left\{\frac{1}{2}\log(1+(1-\rho)\mathsf{SNR}),\right.$$
$$\left.\frac{1}{2}\log\left(\frac{1}{2}+(1+\rho)\mathsf{SNR}\right)\right\}$$

Setting $\rho = \frac{1}{4\mathsf{SNR}}$ we get

$$R < \frac{1}{2}\log\left(\frac{3}{4}+\mathsf{SNR}\right)$$

Summary and Conclusions

- We studied the problem of computing a linear function from the output of a Gaussian MAC with feedback
- We derived a new coding scheme for this scenario
- For a symmetric setting our scheme achieves $R = \frac{1}{2}\log\left(\frac{3}{4} + \mathsf{SNR}\right)$
- The scheme can be extended to noisy feedback and more than 2 users
- Our scheme works in blocks. Can we find a scalar, à la Schalkwijk-Kailath 66 scheme?