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

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

\[ Z \sim \mathcal{N}(0, 1) \]
Gaussian Multiple-Access Channel

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

Capacity Region

\[ R_1 < \frac{1}{2} \log(1 + h_1^2 \text{SNR}) \]
\[ R_2 < \frac{1}{2} \log(1 + h_2^2 \text{SNR}) \]
\[ R_1 + R_2 < \frac{1}{2} \log(1 + \| \mathbf{h} \|^2 \text{SNR}) \]
Gaussian Multiple-Access Channel

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

\[ Z \sim \mathcal{N}(0, 1) \]
Gaussian Multiple-Access Channel

Feedback Capacity Region (Ozarow 84)

\[
R_1 < \frac{1}{2} \log(1 + (1 - \rho^2)h_1^2 \text{SNR})
\]

\[
R_2 < \frac{1}{2} \log(1 + (1 - \rho^2)h_2^2 \text{SNR})
\]

\[
R_1 + R_2 < \frac{1}{2} \log(1 + (\|h\|^2 + 2\rho|h_1h_2|) \text{SNR})
\]
Gaussian MAC - Compute-and-Forward

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

$w_1 \in \mathbb{F}_p^k \rightarrow \mathcal{E}_1 \xrightarrow{} X_1 \xrightarrow{h_1} Y \xrightarrow{} D \xrightarrow{} w_1 \oplus w_2$

$w_2 \in \mathbb{F}_p^k \rightarrow \mathcal{E}_2 \xrightarrow{} X_2 \xrightarrow{h_2} Y$ 

$Z \sim \mathcal{N}(0, 1)$

$Y = h_1 X_1 + h_2 X_2 + Z$
Gaussian MAC - Compute-and-Forward

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

\[ w_1 \in \mathbb{F}_p^k \]

\[ w_2 \in \mathbb{F}_p^k \]

\[ Z \sim \mathcal{N}(0, 1) \]

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

- We assume full CSI everywhere
Gaussian MAC - Compute-and-Forward

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

\[ Z \sim \mathcal{N}(0, 1) \]

\[ Z = \frac{X_1 + X_2 + Z}{\sqrt{\text{SNR}}} \]

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

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

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

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

- At high SNR the bounds coincide. At low SNR separation is optimal
Gaussian MAC - Compute-and-Forward

\[ w_1 \in \mathbb{F}_p^k \]

\[ E_1 \]

\[ X_1 \]

\[ h_1 \]

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

\[ Z \sim \mathcal{N}(0, 1) \]

\[ w_2 \in \mathbb{F}_p^k \]

\[ E_2 \]

\[ X_2 \]

\[ h_2 \]

\[ D \]

\[ w_1 \oplus w_2 \]
Gaussian MAC - Compute-and-Forward

How Much Does Feedback Help?

\[ w_1 \in F^k_p \]

\[ w_2 \in F^k_p \]

\[ X_1, X_2 \]

\[ E_1, E_2 \]

\[ Y = h_1 X_1 + h_2 X_2 + Z \]

\[ Z \sim \mathcal{N}(0, 1) \]

\[ Y \]

\[ w_1 \oplus w_2 \]

Ordentlich, Erez, Nazer

On compute-and-Forward with Feedback
Gaussian MAC - Compute-and-Forward

How Much Does Feedback Help?

- Upper bound remains the same: $C_{\text{comp}} \leq \frac{1}{2} \log (1 + \min\{h_1^2, h_2^2\}\text{SNR})$
Gaussian MAC - Compute-and-Forward

\[ \mathbf{w}_1 \in \mathbb{F}_p^k \rightarrow \mathcal{E}_1 \rightarrow X_1 \rightarrow h_1 \rightarrow Y \]

\[ \mathbf{w}_2 \in \mathbb{F}_p^k \rightarrow \mathcal{E}_2 \rightarrow X_2 \rightarrow h_2 \rightarrow Y = h_1 X_1 + h_2 X_2 + Z \]

\[ Z \sim \mathcal{N}(0, 1) \]

How Much Does Feedback Help?

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

Ordentlich, Erez, Nazer
On compute-and-Forward with Feedback
Gaussian MAC - Compute-and-Forward

How Much Does Feedback Help?

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

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

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

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

Any computation rate satisfying

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

is achievable with feedback.
Gaussian MAC - Compute-and-Forward

$h_1 = 1, \text{SNR}=2$

Rate vs. $h_2$ for $h_1 = 1, \text{SNR}=2$.
Gaussian MAC - Compute-and-Forward

$h_1 = 1, \text{SNR}=2$

Upper bound
Computation rate - full CSI

Ordentlich, Erez, Nazer
On compute-and-Forward with Feedback
Gaussian MAC - Compute-and-Forward

$h_1=1, \text{SNR}=2$

Rate vs. $h_2$

- **Upper bound**
- **Computation rate - full CSI**
- **Separation with feedback**

Ordentlich, Erez, Nazer

On compute-and-Forward with Feedback
Gaussian MAC - Compute-and-Forward

$h_1 = 1, \text{SNR}=2$

Rate

$h_2$

Upper bound
Computation rate - full CSI
Separation with feedback
Feedback computation rate

Ordentlich, Erez, Nazer
On compute-and-Forward with Feedback
Lattice List Decoder (Song & Devroye 13)

Fine lattice \( \Lambda_c \)
Fine lattice $\Lambda_c$, coarse lattice $\Lambda \subseteq \Lambda_c$

$$\mathcal{C} = \Lambda_c \cap \mathcal{V}$$
Fine lattice $\Lambda_c$, coarse lattice $\Lambda$, intermediate lattice $\Lambda_s$, $\Lambda \subseteq \Lambda_s \subseteq \Lambda_c$

$\mathcal{C} = \Lambda_c \cap \mathcal{V}$
Lattice List Decoder (Song & Devroye 13)

AWGN channel $y = x + z$, $R > C$
Lattice List Decoder (Song & Devroye 13)

AWGN channel $y = x + z$, $R > C$
AWGN channel $y = x + z$, $R > C$
Lattice List Decoder (Song & Devroye 13)

\[ y = x + z, \quad R > C \]
Lattice List Decoder (Song & Devroye 13)

AWGN channel $y = x + z$, $R > C$

Decode a list of codewords: $L = \{c \in C : c \in [y + V_s] \mod \Lambda\}$
Lattice List Decoder (Song & Devroye 13)

AWGN channel \( y = x + z, \quad R > C \)

Decode a list of codewords: \( L = \{ c \in C : c \in [y + V_s] \mod \Lambda \} \)

\(|L| = \log \left( \frac{\text{Vol}(V_s)}{\text{Vol}(V_c)} \right)\)
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.
High-level overview of our coding scheme
High-level overview of our coding scheme

- Block Markov coding
High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user $i$ can decode $w_i$ using the feedback link
High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user $i$ can decode $w_i$ 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*
High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user $i$ can decode $w_i$ 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
High-level overview of our coding scheme

- Block Markov coding
- In the end of each block user $i$ can decode $\mathbf{w}_i$ 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 \textit{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 $w_1^{(k)} \oplus w_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
For simplicity assume \( h_1 = h_2 = 1 \)

- Decoding \( w_1^{(k)} \oplus w_2^{(k)} \), \( k = 1, \ldots, N \) over \( N + 1 \) blocks
- Both users encode their messages using the same lattice code \( C \), such that \( \tilde{x}_i^{(k)} = f \left( w_i^{(k)} \right) \in C \)
Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $w_1^{(k)} \oplus w_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $C$, such that $\tilde{x}_i^{(k)} = f \left( w_i^{(k)} \right) \in C$
- First block: $x_i^{(1)} = \sqrt{1 - \rho} \tilde{x}_i^{(1)}$
For simplicity assume $h_1 = h_2 = 1$

- Decoding $w_1^{(k)} \oplus w_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $C$, such that $\tilde{x}_i^{(k)} = f \left( w_i^{(k)} \right) \in C$
- First block: $x_i^{(1)} = \sqrt{1 - \rho \tilde{x}_i^{(1)}}$
- Receiver sees

$$y^{(1)} = \sqrt{1 - \rho \left( \tilde{x}_1^{(1)} + \tilde{x}_2^{(1)} \right)} + z^{(1)}.$$
Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $w_1^{(k)} \oplus w_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $C$, such that $\tilde{x}_i^{(k)} = f\left(w_i^{(k)}\right) \in C$
- First block: $x_i^{(1)} = \sqrt{1 - \rho} \tilde{x}_i^{(1)}$
- Receiver sees

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

- $R < R_{\text{comp}} \triangleq \frac{1}{2} \log \left(\frac{1}{2} + (1 - \rho)\text{SNR}\right)$ is needed for decoding $w_1^{(1)} \oplus w_2^{(1)}$.
Compute-and-Forward with Feedback

For simplicity assume $h_1 = h_2 = 1$

- Decoding $w_1^{(k)} \oplus w_2^{(k)}$, $k = 1, \ldots, N$ over $N + 1$ blocks
- Both users encode their messages using the same lattice code $C$, such that $\tilde{x}_i^{(k)} = f \left( w_i^{(k)} \right) \in C$
- First block: $x_i^{(1)} = \sqrt{1 - \rho} \tilde{x}_i^{(1)}$
- Receiver sees

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

- $R < R_{\text{comp}} \triangleq \frac{1}{2} \log \left( \frac{1}{2} + (1 - \rho) \text{SNR} \right)$ is needed for decoding $w_1^{(1)} \oplus w_2^{(1)}$
- In our case $R > R_{\text{comp}}$ and the receiver can decode a list $L^{(1)}$ of candidates for $w_1^{(1)} \oplus w_2^{(1)}$ with size $|L^{(1)}| = 2^{n(R - R_{\text{comp}})}$. 

Ordentlich, Erez, Nazer
Compute-and-Forward with Feedback

- Using the feedback link, user $i$ can decode $w_i^{(1)}$ if

\[
R < \frac{1}{2} \log(1 + (1 - \rho)\text{SNR})
\]
Using the feedback link, user $i$ can decode $\mathbf{w}_i^{(1)}$ if

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

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 $w_i^{(1)}$ if
  \[ R < \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}) \]

- Both users can compute $v^{(1)} = w_1^{(1)} \oplus w_2^{(1)}$

- Both users apply the same binning function $B : [2^{nR}] \mapsto [2^{nR'}], R' < R$, to obtain $B(v^{(1)})$
Compute-and-Forward with Feedback

- Using the feedback link, user $i$ can decode $w_i^{(1)}$ if

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

- Both users can compute $v^{(1)} = w_1^{(1)} \oplus w_2^{(1)}$
- Both users apply the same binning function $B : [2^{nR}] \mapsto [2^{nR'}], R' < R$, to obtain $B(v^{(1)})$
- Each user encodes $B(v^{(1)})$ to $x_{\text{cohr}}^{(1)}$ using the same codebook $C'$ with rate $R'$ and average power $\text{SNR}$
Compute-and-Forward with Feedback

Using the feedback link, user $i$ can decode $w_i^{(1)}$ if

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

Both users can compute $v^{(1)} = w_1^{(1)} \oplus w_2^{(1)}$

Both users apply the same binning function $B : [2^{nR}] \mapsto [2^{nR'}]$, $R' < R$, to obtain $B(v^{(1)})$

Each user encodes $B(v^{(1)})$ to $x^{(1)}_{\text{cohr}}$ using the same codebook $C'$ with rate $R'$ and average power SNR.

In addition, each user encodes a new message $w_i^{(2)}$ to the codeword $\tilde{x}_i^{(2)}$ and transmits

$$x_i^{(2)} = \sqrt{\rho}x_{\text{cohr}}^{(1)} + \sqrt{1 - \rho}\tilde{x}_i^{(2)}$$
Compute-and-Forward with Feedback

- Channel output is

\[ y^{(2)} = 2\sqrt{\rho}x^{(1)}_{\text{cohr}} + \sqrt{1 - \rho} \left( \tilde{x}_1^{(2)} + \tilde{x}_2^{(2)} \right) + z^{(2)} \]
Compute-and-Forward with Feedback

- Channel output is

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

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

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

- Can decode \( x^{(1)}_{\text{cohr}} \) 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 \( w \in \mathbb{F}_p^k \) in \( L^{(1)} \cap B^{-1}(v^{(1)}) \)
Compute-and-Forward with Feedback

- Channel output is

\[ y^{(2)} = 2\sqrt{\rho}x^{(1)}_{\text{coh}} + \sqrt{1-\rho} (\tilde{x}_1^{(2)} + \tilde{x}_2^{(2)}) + z^{(2)} \]

- Can decode \( x^{(1)}_{\text{coh}} \) 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 \( w \in \mathbb{F}_p^k \) in \( L^{(1)} \cap B^{-1}(v^{(1)}) \)

- If \( R' > R - R_{\text{comp}} \) such a \( w \in \mathbb{F}_p^k \) will be found with probability 1
Compute-and-Forward with Feedback

- Channel output is

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

- Can decode \( x^{(1)}_{\text{cohr}} \) 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 \( w \in \mathbb{F}_p^k \) in \( L^{(1)} \cap B^{-1}(v^{(1)}) \)

- If \( R' > R - R_{\text{comp}} \) such a \( w \in \mathbb{F}_p^k \) will be found with probability 1

- Next, the decoder subtracts \( x^{(1)}_{\text{cohr}} \) from \( y^{(2)} \) and decodes a list \( L^{(2)} \) of candidates for \( v^{(2)} = w^{(2)}_1 \oplus w^{(2)}_2 \)
Correct decoding through feedback link

\[ R < \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}) \]
Correct decoding through feedback link

\[ R < \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}) \]

Correct decoding of \( x_{\text{cohr}} \)

\[ R' \leq \frac{1}{2} \log \left( 1 + \frac{4\rho\text{SNR}}{1 + 2(1 - \rho)\text{SNR}} \right) \]
Compute-and-Forward with Feedback - Rate Constraints

- Correct decoding through feedback link

\[ R < \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}) \]

- Correct decoding of \( x_{\text{cohr}} \)

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

- Unique element in intersection of list and bin

\[ R' > R - \frac{1}{2} \log \left( \frac{1}{2} + (1 - \rho)\text{SNR} \right) \]
Achievable Rate

\[ R < \min \left\{ \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}), \right. \]
\[ \left. \frac{1}{2} \log \left( 1 + \frac{4\rho \text{SNR}}{1 + 2(1 - \rho)\text{SNR}} \right) + \frac{1}{2} \log \left( \frac{1}{2} + (1 - \rho)\text{SNR} \right) \right\} \]
Achievable Rate

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

\[ \left. \frac{1}{2} \log \left( \frac{1}{2} + (1 + \rho)SNR \right) \right\} \]
Achievable Rate

\[ R < \min \left\{ \frac{1}{2} \log(1 + (1 - \rho)\text{SNR}), \quad \frac{1}{2} \log \left( \frac{1}{2} + (1 + \rho)\text{SNR} \right) \right\} \]

Setting \( \rho = \frac{1}{4\text{SNR}} \) we get

\[ R < \frac{1}{2} \log \left( \frac{3}{4} + \text{SNR} \right) \]
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 $R = \frac{1}{2} \log \left( \frac{3}{4} + \text{SNR} \right)$.
- 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?