Hybrid Coding

Young-Han Kim, UCSD

Living Information Theory Workshop

Happy $<70.42>^{th}$ birthday, Professor Berger!
Hybrid Coding

Young-Han Kim, UCSD

Living Information Theory Workshop

Happy $<70.42>$th birthday, Professor Berger!

Sung Hoon Lim, Paolo Minero, and Abbas El Gamal
Gratitude #1

INFORMATION AND CONTROL 13, 254–273 (1968)

Rate Distortion Theory for Sources with Abstract Alphabets and Memory

Toby Berger†

Advanced Development Laboratory, Raytheon Company
Wayland, Massachusetts 01778
Rate Distortion Theory for Sources with Abstract Alphabets and Memory

Toby Berger†

Advanced Development Laboratory, Raytheon Company
Wayland, Massachusetts 01778

**Definition 1.** A source \([X, \mu]\) is block ergodic if it is \(\tau\)-ergodic for every positive \(\tau \in M\).

We show in the appendix that block ergodicity lies between ergodicity and weak mixing in restrictiveness. Note that \(\tau\)-ergodicity of \([X, \mu]\) and ergodicity of \([X, \mu], \tau\) are equivalent.
Gratitude #2
Gratitude #2

- Strong typicality
- (Extended) Markov lemma
Gratitude #3
Gratitude #3
But life is short and information endless: nobody has time for everything. In practice we are generally forced to choose between an unduly brief exposition and no exposition at all. Abbreviation is a necessary evil and the abbreviator’s business is to make the best of a job which, though intrinsically bad, is still better than nothing. He must learn to simplify, but not to the point of falsification. He must learn to concentrate upon the essentials of a situation, but without ignoring too many of reality’s qualifying side issues.

—Aldous Huxley

Brave New World Revisited
Point-to-Point Communication System

Question

Find the sufficient and necessary condition to achieve

$$\limsup_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} d(S_i, \hat{S}_i) \leq D$$
Question

Find the sufficient and necessary condition to achieve

\[
\limsup_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} d(S_i, \hat{S}_i) \leq D
\]

Answer: Source–channel separation (Shannon 1948, 1959)

• Sufficient and necessary condition:

\[
R(D) = \min_{p(\hat{S}|S): E(d(S,\hat{S})) \leq D} I(S; \hat{S}) < \max_{p(x)} I(X; Y) = C
\]
Question

Find the sufficient and necessary condition to achieve

$$\limsup_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} d(S_i, \hat{S}_i) \leq D$$

Answer: Source–channel separation (Shannon 1948, 1959)

- Sufficient and necessary condition:

$$R(D) = \min_{p(\hat{S}|S) : \mathbb{E}(d(S, \hat{S})) \leq D} I(S; \hat{S}) \leq \max_{p(x)} I(X; Y) = C$$
Find the sufficient and necessary condition to achieve

$$\limsup_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} d(S_i, \hat{S}_i) \leq D$$

Answer: Source–channel separation (Shannon 1948, 1959)

- Sufficient and necessary condition:

$$R(D) = \min_{p(\hat{s}|s): E(d(S,\hat{S})) \leq D} I(S; \hat{S}) \leq \max_{p(x)} I(X; Y) = C$$

- Digital source–channel interface
Uncoded Transmission

\[ S^n \rightarrow \text{Source–channel encoder} \rightarrow X^n \rightarrow p(y|x) \rightarrow Y^n \rightarrow \text{Source–channel decoder} \rightarrow \hat{S}^n \]
Uncoded Transmission

\[ S^n \rightarrow x(s) \rightarrow X^n \rightarrow p(y|x) \rightarrow \gamma^n \rightarrow \hat{s}(y) \rightarrow \hat{S}^n \]
Uncoded Transmission

- Optimal for quadratic Gaussian source coding over the Gaussian channel with average power constraint (Goblick 1965)
Uncoded Transmission

- Optimal for quadratic Gaussian source coding over the Gaussian channel with average power constraint (Goblick 1965)

\[ R(D) = C \]
Uncoded Transmission

- Optimal for quadratic Gaussian source coding over the Gaussian channel with average power constraint (Goblick 1965)

\[ R(D) = C \]

- “To code or not to code” (Gastpar, Rimoldi, and Vetterli 2003)
Optimal for quadratic Gaussian source coding over the Gaussian channel with average power constraint (Goblick 1965)

\[ R(D) = C \]

“To code or not to code” (Gastpar, Rimoldi, and Vetterli 2003)

Analog source–channel interface
Hybrid Coding

\[ S^n \xrightarrow{\text{Source–channel encoder}} X^n \xrightarrow{p(y|x)} Y^n \xrightarrow{\text{Source–channel decoder}} \hat{S}^n \]
Hybrid Coding

\[ S^n \xrightarrow{U^n(M)} x(u,s) \xrightarrow{p(y|x)} Y^n \xrightarrow{U^n(\hat{M})} \hat{s}(u,y) \Rightarrow \hat{S}^n \]
Hybrid Coding

\[ \begin{align*} \mathcal{S}^n &\xrightarrow{\text{Source encoder}} \mathcal{U}^n(M) \xrightarrow{x(u,s)} \mathcal{X}^n \xrightarrow{p(y|x)} \mathcal{Y}^n \xrightarrow{\text{Channel decoder}} \hat{\mathcal{U}}^n(M) \xrightarrow{\hat{s}(u,y)} \hat{\mathcal{S}}^n \end{align*} \]

- Sufficient condition:
  \[ I(U;S) < I(U;Y) \]

  for some \( p(u|s), x(u,s), \hat{s}(u,y) \) such that \( E(d(S,\hat{S})) \leq D \)
Hybrid Coding

- Sufficient condition:
  \[ I(U; S) < I(U; Y) \]
  for some \( p(u|s), x(u, s), \hat{s}(u, y) \) such that \( E(d(S, \hat{S})) \leq D \)
- Analog/digital hybrid source–channel interface
Hybrid Coding

- Sufficient condition:
  \[ I(U; S) < I(U; Y) \]
  for some \( p(u|s), x(u, s), \hat{s}(u, y) \) such that \( E(d(S, \hat{S})) \leq D \)

- Analog/digital hybrid source–channel interface
  - \( U = \emptyset \): uncoded transmission
Hybrid Coding

- Sufficient condition:
  \[ I(U; S) < I(U; Y) \]
  for some \( p(u|s), x(u, s), \hat{s}(u, y) \) such that \( E(d(S, \hat{S})) \leq D \)

- Analog/digital hybrid source–channel interface
  - \( U = \emptyset \): uncoded transmission
  - \( U = (\hat{S}, X) \sim p(\hat{s}|s)p(x) \): separate source and channel coding
Hybrid Coding

- Sufficient condition:
  \[ I(U; S) < I(U; Y) \]
  for some \( p(u|s), x(u, s), \hat{s}(u, y) \) such that \( E(d(S, \hat{S})) \leq D \)

- Analog/digital hybrid source–channel interface
  - \( U = \emptyset \): uncoded transmission
  - \( U = (\hat{S}, X) \sim p(\hat{s}|s)p(x) \): separate source and channel coding

Hybrid Coding

- **Sufficient condition:**
  \[ I(U; S) < I(U; Y) \]
  for some \( p(u|s), x(u,s), \hat{s}(u,y) \) such that \( E(d(S, \hat{S})) \leq D \)

- **Analog/digital** hybrid source–channel interface
  - \( U = \emptyset \): uncoded transmission
  - \( U = (\hat{S}, X) \sim p(\hat{s}|s)p(x) \): separate source and channel coding

Proof of Achievability

\[ S^n \xrightarrow{U^n(M)} x(u,s) \xrightarrow{X^n} p(y|x) \xrightarrow{Y^n} \hat{s}(u,y) \xrightarrow{\hat{S}^n} \]

Source encoder
Channel decoder

\[ \hat{s}(u,y) \]
Proof of Achievability

Source encoder $U^n(M)$

Random codebook generation
$2^{nR}$ sequences $u^n(m) \sim \prod_{i=1}^n p_U(u_i)$ for $m \in [1:2^{nR}]$
Proof of Achievability

Random codebook generation

$2^{nR}$ sequences $u^n(m) \sim \prod_{i=1}^{n} p_{U}(u_i)$ for $m \in [1:2^{nR}]$

Joint typicality encoding

Find an index $m$ such that $(u^n(m), s^n) \in \mathcal{T}^{(n)}_{\epsilon'}$ and transmit $x_i = x(u_i(m), s_i)$:
Proof of Achievability

Random codebook generation

$2^{nR}$ sequences $u^n(m) \sim \prod_{i=1}^{n} p_U(u_i)$ for $m \in [1:2^{nR}]$

Joint typicality encoding

Find an index $m$ such that $(u^n(m), s^n) \in \mathcal{T}_{\epsilon'}^{(n)}$ and transmit $x_i = x(u_i(m), s_i)$: successful w.h.p. if $R > I(U; S)$
Proof of Achievability

Random codebook generation

$2^{nR}$ sequences $u^n(m) \sim \prod_{i=1}^n p_U(u_i)$ for $m \in [1:2^{nR}]$

Joint typicality encoding

Find an index $m$ such that $(u^n(m), s^n) \in T_{\epsilon}^{(n)}$ and transmit $x_i = x(u_i(m), s_i)$: successful w.h.p. if $R > I(U;S)$

Joint typicality decoding

Find the unique index $\hat{m}$ such that $(u^n(\hat{m}), y^n) \in T_{\epsilon}^{(n)}$ and reconstruct $\hat{s}_i = \hat{s}(u_i(\hat{m}), y_i)$:
Proof of Achievability

**Random codebook generation**

$2^{nR}$ sequences $u^n(m) \sim \prod_{i=1}^n p_{U}(u_i)$ for $m \in [1:2^{nR}]$

**Joint typicality encoding**

Find an index $m$ such that $(u^n(m), s^n) \in T_{\epsilon'}(n)$ and transmit $x_i = x(u_i(m), s_i)$: successful w.h.p. if $R > I(U; S)$

**Joint typicality decoding**

Find the unique index $\hat{m}$ such that $(u^n(\hat{m}), y^n) \in T_{\epsilon}(n)$ and reconstruct $\hat{s}_i = \hat{s}(u_i(\hat{m}), y_i)$: successful w.h.p. if $R < I(U; Y)$
Correlated Sources over a MAC

Channel encoder 1

Source encoder 2

\( S^n_1 \)

\( U^n_1(M_1) \)

\( X^n_1 \)

\( p(y|x_1, x_2) \)

\( Y^n \)

\( U^n_1(\hat{M}_1) \)

\( \hat{s}_1(u_1, u_2, y) \)

\( \hat{S}^n_1 \)

Hybrid coding

- Berger–Tung source coding + MAC coding

\( S^n_2 \)

\( U^n_2(M_2) \)

\( X^n_2 \)

\( \hat{s}_2(u_1, u_2, y) \)

\( \hat{S}^n_2 \)
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding

\[ R_1 > I(U_1; S_1), \]
\[ R_2 > I(U_2; S_2) \]
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding

\[ R_1 > I(U_1; S_1), \]
\[ R_2 > I(U_2; S_2) \]

\[ R_1 < I(U_1; Y, U_2), \]
\[ R_2 < I(U_2; Y, U_1), \]
\[ R_1 + R_2 < I(U_1, U_2; Y) + I(U_1; U_2) \]
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding

\[ I(U_1; S_1|Q) < I(U_1; Y, U_2|Q), \]
\[ I(U_2; S_2|Q) < I(U_2; Y, U_1|Q), \]
\[ I(U_1; S_1|Q) + I(U_2; S_2|Q) < I(U_1, U_2; Y|Q) + I(U_1; U_2|Q) \]

for some \( p(q)p(u_1|s_1, q)p(u_2|s_2, q), x_j(u_j, s_j, q), \hat{s}_j(u_1, u_2, y, q) \) such that \( E(d_j(S_j, \hat{S}_j)) \leq D_j, j = 1, 2 \)
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding
Correlated Sources over a MAC

Hybrid coding

• Berger–Tung source coding + MAC coding

• Special cases
  • $(\hat{S}_1, \hat{S}_2) = (S_1, S_2)$: lossless coding (Cover, El Gamal, and Salehi 1980)
Hybrid coding

- Berger–Tung source coding + MAC coding

- Special cases
  - \((\hat{S}_1, \hat{S}_2) = (S_1, S_2)\): lossless coding (Cover, El Gamal, and Salehi 1980)
  - \(Y = (X_1, X_2)\): noiseless channel (Berger 1978, Tung 1978)
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding

- Special cases
  - \((\hat{S}_1, \hat{S}_2) = (S_1, S_2)\): lossless coding (Cover, El Gamal, and Salehi 1980)
  - \(Y = (X_1, X_2)\): noiseless channel (Berger 1978, Tung 1978)
  - Gaussian \((S_1, S_2)\) and Gaussian MAC (Tinguely and Lapidoth 2006)
Correlated Sources over a MAC

Hybrid coding

- Berger–Tung source coding + MAC coding

- Special cases
  - \((\hat{S}_1, \hat{S}_2) = (S_1, S_2)\): lossless coding (Cover, El Gamal, and Salehi 1980)
  - \(Y = (X_1, X_2)\): noiseless channel (Berger 1978, Tung 1978)
  - Gaussian \((S_1, S_2)\) and Gaussian MAC (Tinguely and Lapidoth 2006)

- Common part: Kaspi–Berger coding + Slepian–Wolf MAC coding
2-Sender 2-Receiver Channel

\[ S_1^n \rightarrow \text{Encoder 1} \rightarrow X_1^n \rightarrow p(y_1, y_2 | x_1, x_2) \rightarrow Y_1^n \rightarrow \text{Decoder 1} \rightarrow \hat{S}_{11}^n, \hat{S}_{21}^n \]

\[ S_2^n \rightarrow \text{Encoder 2} \rightarrow X_2^n \rightarrow Y_2^n \rightarrow \hat{S}_{12}^n, \hat{S}_{22}^n \]
2-Sender 2-Receiver Channel

Hybrid coding
2-Sender 2-Receiver Channel

Hybrid coding

- Conceptually simple yet performs very well
2-Sender 2-Receiver Channel

Hybrid coding

- Conceptually simple yet performs very well
- Paolo Minero’s talk on Monday 10:05 am (right before David Tse)
Diamond Network

$M \xrightarrow{\text{Encoder}} X_1^n \xrightarrow{p(y_2, y_3|x_1)} Y_2^n \xrightarrow{\text{Relay 2}} X_2^n \xrightarrow{p(y_4|x_2, x_3)} Y_4^n \xrightarrow{\text{Decoder}} \hat{M}$

Existing coding schemes

- **Decode–forward (digital-to-digital interface)**
Diamond Network

Existing coding schemes

- **Decode–forward** (digital-to-digital interface)
- **Amplify–forward** (analog-to-analog interface)
Existing coding schemes

- **Decode–forward** (digital-to-digital interface)
- **Amplify–forward** (analog-to-analog interface)
- **Noisy network coding** (analog-to-digital interface)
Diamond Network

Existing coding schemes

- Decode–forward (digital-to-digital interface)
- Amplify–forward (analog-to-analog interface)
- Noisy network coding (analog-to-digital interface)

Question

Can we do better than these coding schemes?
Diamond Network

\[ M \rightarrow \text{Encoder} \xrightarrow{X_1^n} p(y_2, y_3 | x_1) \xrightarrow{Y_2^n} \text{Relay 2} \xrightarrow{X_2^n} p(y_4 | x_2, x_3) \xrightarrow{Y_4^n} \text{Decoder} \rightarrow \hat{M} \]

**Answer**

- Hybrid coding *(analog-to-[analog/digital] interface)*
Answer

- Hybrid coding (analog-to-[analog/digital] interface)
Diamond Network

Answer

- **Hybrid coding** *(analog-to-[analog/digital] interface)*

- Naturally combines noisy network coding and amplify-forward
Example 1: Deterministic Diamond Network

Encoder $X^n_1 \rightarrow \begin{pmatrix} y_2, y_3 \end{pmatrix}(x_1)\rightarrow Relay 2 \rightarrow \begin{pmatrix} y_4 \end{pmatrix}(x_2, x_3) \rightarrow Decoder \rightarrow \hat{M}$
Example 1: Deterministic Diamond Network

Cutset bound

\[ C \leq \max_{p(x_1,x_2,x_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \} \]
Example 1: Deterministic Diamond Network

\[ M \rightarrow \text{Encoder} \rightarrow (y_2, y_3)(x_1) \rightarrow \text{Relay 2} \rightarrow X_2^n \rightarrow \text{Relay 3} \rightarrow X_3^n \rightarrow \text{Decoder} \rightarrow \hat{M} \]

**Cutset bound**

\[ C \leq \max_{p(x_1, x_2, x_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \} \]

**General lower bound (Avestimehr, Diggavi, and Tse 2007)**

\[ C \geq \max_{p(x_1)p(x_2)p(x_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \} \]
Example 1: Deterministic Diamond Network

\[ M \rightarrow \text{Encoder} \rightarrow \begin{pmatrix} (y_2, y_3)(x_1) \end{pmatrix} \rightarrow \text{Relay 2} \rightarrow \begin{pmatrix} y_4(x_2, x_3) \end{pmatrix} \rightarrow \text{Decoder} \rightarrow \hat{M} \]

**Cutset bound**

\[
C \leq \max_{p(x_1, x_2, x_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \}
\]

**General lower bound (Avestimehr, Diggavi, and Tse 2007)**

\[
C \geq \max_{p(x_1)p(x_2)p(x_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \}
\]

**Hybrid coding lower bound**

\[
C \geq \max_{p(x_1)p(x_2|y_2)p(x_3|y_3)} \min \{ H(Y_2, Y_3), H(Y_2) + H(Y_4 | X_3), H(Y_3) + H(Y_4 | X_2), H(Y_4) \}
\]
Example 1: Deterministic Diamond Network

\[ M \xrightarrow[]{} \text{Encoder} \xrightarrow[]{} (y_2, y_3)(x_1) \xrightarrow[]{} \text{Relay 2} \xrightarrow[]{} y_4(x_2, x_3) \xrightarrow[]{} \text{Decoder} \xrightarrow[]{} \hat{M} \]

**Blackwell BC** \((y_2(x_1), y_3(x_1))\)

\[
\begin{array}{ccc}
X_1 & \rightarrow & 0 \\
& & 0 \rightarrow Y_2 \\
& & 1 \rightarrow Y_3 \\
2 & \rightarrow & 0 \\
& & 1 \\
\end{array}
\]

**Binary erasure MAC** \(y_4(x_2, x_3)\)

\[
Y_4 = X_2 + X_3
\]

\[X_2, X_3 \in \{0,1\}, \quad Y_4 \in \{0,1,2\}\]
Example 1: Deterministic Diamond Network

\[ M \xrightarrow{\text{Encoder}} X_1^n \xrightarrow{(y_2, y_3)(x_1)} Y_2^n \xrightarrow{\text{Relay 2}} X_2^n \xrightarrow{(y_4(x_2, x_3))} Y_4^n \xrightarrow{\text{Decoder}} \hat{M} \]

Blackwell BC \((y_2(x_1), y_3(x_1))\)

<table>
<thead>
<tr>
<th>X_1</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ Y_2 = \begin{cases} 0 & \text{if } X_1 = 0 \\ 1 & \text{if } X_1 = 1 \end{cases} \]

Binary erasure MAC \(y_4(x_2, x_3)\)

\[ Y_4 = X_2 + X_3 \]

\[ X_2, X_3 \in \{0, 1\}, \quad Y_4 \in \{0, 1, 2\} \]

General lower bound = \(1.5 < \log 3\) = Hybrid coding lower bound
Example 2: Gaussian Diamond Network
Example 2: Gaussian Diamond Network

\[ X \xrightarrow{g} Z_1 \xrightarrow{Y_1} X_2 \xrightarrow{g} Z_3 \xrightarrow{Y_3} X_3 \xrightarrow{g} Z_4 \xrightarrow{Y_4} \]
AF vs. NNC vs. Hybrid Coding

The graph illustrates the performance comparison between AF, NNC, and Hybrid Coding with respect to the parameter $g$. The x-axis represents $g$, ranging from 0 to 30, while the y-axis shows the performance metric ranging from 0.35 to 0.8. The curves for NNC, AF, and Hybrid Coding are indicated by different colors and line styles for easy differentiation.
AF vs. NNC vs. Hybrid Coding
Example 3: Gaussian Two-Way Relay Channel
Example 3: Gaussian Two-Way Relay Channel

\[ r^{-3/2} \quad (1 - r)^{-3/2} \]

\[ \begin{align*}
X_1 & \rightarrow Z_3 & \leftarrow X_2 \\
& \downarrow r^{-3/2} & \uparrow (1 - r)^{-3/2} \\
& \downarrow & \\
& \downarrow & \\
Y_3 & \leftarrow & \\
& \downarrow & \\
X_3 & \leftarrow & \\
& \downarrow r^{-3/2} & \uparrow (1 - r)^{-3/2} \\
Y_1 & \rightarrow Z_1 & \leftarrow Y_2 \\
& \downarrow & \\
& \downarrow & \\
Z_1 & \leftarrow & \\
& \downarrow & \\
& \downarrow & \\
Z_2 & \leftarrow & \\
& \downarrow & \\
& \downarrow & \\
& \downarrow & \\
1 & \rightarrow & 3 \\
& \downarrow & \leftarrow \\
& \downarrow & \leftarrow \\
1 & \rightarrow 2 \end{align*} \]
Hybrid Coding vs. Other Coding Schemes

![Graph showing the comparison between Hybrid Coding and other coding schemes. The graph plots the rate $R$ as a function of $r$. The curves represent $R_{CS}$, $R_{DF}$, $R_{AF}$, and $R_{NNC}$.](image-url)
Hybrid Coding vs. Other Coding Schemes

The diagram compares the performance of different coding schemes. The axes represent a variable $r$ and a range of values from 6.5 to 2.5. Several curves are plotted, each labeled with $R_{CS}$, $R_{HC}$, $R_{DF}$, $R_{AF}$, and $R_{NNC}$. These curves illustrate the relationship between $r$ and the performance metrics for each coding scheme.
Concluding Remarks

Hybrid coding
Concluding Remarks

Hybrid coding

• Versatile interface for source–channel coding and relaying
Concluding Remarks

## Hybrid coding

- Versatile interface for *source–channel coding* and *relaying*
- Joint source–channel coding can be *simple* (and *fun* too!)
Hybrid coding

- Versatile interface for source–channel coding and relaying
- Joint source–channel coding can be simple (and fun too!)
- Hybrid coding > amplify–forward + noisy network coding
Concluding Remarks

Hybrid coding

- Versatile interface for source–channel coding and relaying
- Joint source–channel coding can be simple (and fun too!)
- Hybrid coding > amplify–forward + noisy network coding
- Hybrid coding + structured coding > decode–forward
Concluding Remarks

Hybrid coding

• Versatile interface for source–channel coding and relaying

• Joint source–channel coding can be simple (and fun too!)

• Hybrid coding > amplify–forward + noisy network coding

• Hybrid coding + structured coding > decode–forward

Back to reality
Concluding Remarks

Hybrid coding

• Versatile interface for source–channel coding and relaying

• Joint source–channel coding can be simple (and fun too!)

• Hybrid coding > amplify–forward + noisy network coding

• Hybrid coding + structured coding > decode–forward

Back to reality

Concluding Remarks

Hybrid coding

- Versatile interface for source–channel coding and relaying
- Joint source–channel coding can be simple (and fun too!)
- Hybrid coding > amplify–forward + noisy network coding
- Hybrid coding + structured coding > decode–forward

Back to reality

- Meir Feder: Amimon high-definition wireless audio-video modem
Concluding Remarks

Hybrid coding

- Versatile interface for source–channel coding and relaying
- Joint source–channel coding can be simple (and fun too!)
- Hybrid coding > amplify–forward + noisy network coding
- Hybrid coding + structured coding > decode–forward

Back to reality

- Meir Feder: Amimon high-definition wireless audio-video modem
- Any good code for both (source) encoding and (channel) decoding?