LDPC Code Design for Transmission of Correlated Sources Across Noisy Channels Without CSIT

Henry Pfister
(Joint work with Arvind Yedla & Krishna Narayanan)

Texas A&M University

ISTC 2010
Brest, France
The Sensor Reachback Problem

[BS03]
The Sensor Reachback Problem

[sources]

Design of Universal LDPC Codes
WCL, Texas A&M University
The Sensor Reachback Problem

i.i.d. draws from $S$ correlated sources $\sim \Pr(X_1, X_2, \cdots, X_S)$ with independent encoding

[BS03]
The Sensor Reachback Problem

$S$ independent memoryless channels

$$\Pr(Y_1^S \mid X_1^S) = \prod_i \Pr(Y_i \mid X_i)$$

i.i.d. draws from $S$ correlated sources $\sim \Pr(X_1, X_2, \ldots, X_S)$ with independent encoding

[BS03]
The Sensor Reachback Problem

$S$ independent memoryless channels

$\Pr(Y_1^S \mid X_1^S) = \prod_i \Pr(Y_i \mid X_i)$

i.i.d. draws from $S$ correlated sources $\sim \Pr(X_1, X_2, \cdots, X_S)$

with independent encoding

$[BS03]$
A Simplified Model

Correlated Sources

Source 1 → Encoder 1 → Channel 1 → Decoder

Source 2 → Encoder 2 → Channel 2

Each source is characterized by a single parameter, not known at the transmitter.

Each code has rate $R$. 

$x_1$ and $y_1$ are outputs of Channel 1.

$x_2$ and $y_2$ are outputs of Channel 2.
A Simplified Model

- each capacity characterized by a single parameter $\alpha$
A Simplified Model

- each capacity characterized by a single parameter $\alpha$
- $\alpha_1$ and $\alpha_2$ not known at transmitter
A Simplified Model

- each capacity characterized by a single parameter $\alpha$
- $\alpha_1$ and $\alpha_2$ not known at transmitter
- each code has rate $R$
Slepian-Wolf Conditions

$\alpha_2$

$(\alpha_1, \alpha_2)$-plane

$\alpha_1$
Slepian-Wolf Conditions

\[ \frac{C_2(\alpha_2)}{R_2} \geq H(U_2|U_1) \]

\[ \frac{C_1(\alpha_1)}{R_1} + \frac{C_2(\alpha_2)}{R_2} \geq H(U_1, U_2) \]

\[ \frac{C_1(\alpha_1)}{R_1} \geq H(U_1|U_2) \]

illustrated for the BEC case where \( C(\alpha) = 1 - \alpha \)
Slepian-Wolf Conditions

\[
\frac{C_2(\alpha_2)}{R_2} \geq H(U_2|U_1)
\]

\[
\frac{C_1(\alpha_1)}{R_1} + \frac{C_2(\alpha_2)}{R_2} \geq H(U_1, U_2)
\]

\[
\frac{C_1(\alpha_1)}{R_1} \geq H(U_1|U_2)
\]

SW channel parameter region
Slepian-Wolf Conditions

\[ 1 - H(U_2 | U_1) R_2 \]

\[ 1 - H(U_1) R_1 \]

\[ 1 - H(U_1 | U_2) R_1 \]

symmetric channel condition
Slepian-Wolf Conditions

\[ 1 - H(U_2 | U_1) R_2 \]

\[ 1 - H(U_2) R_2 \]

symmetric channel condition

achievable channel parameter region (ACPR) of a code optimized for the symmetric channel condition
Slepian-Wolf Conditions

what if the channel condition happened to be different?
Slepian-Wolf Conditions

\[ 1 - H(U_2 | U_1) R_2 \]

\[ 1 - H(U_2) R_2 \]

the performance is bad.
Slepian-Wolf Conditions

\[1 \! - \! H(U_2 | U_1) R_2 \]

\[1 \! - \! H(U_2) R_2 \]

need convergence on the dominant face
Slepian-Wolf Conditions

universal codes

ACPR = full SW region
random codes with ML decoding are universal.
what about LDPC codes with iterative decoding?
Prior Work and Summary

- Prior Work

Optimized LT codes are not universal [YPN09]
carefully designed turbo codes have large ACPRs [AFMFR09]
systematic LDPC codes perform poorly [MFAFR10]

Summary of this work

systematic codes ⇒ correlated codes ⇒ suboptimal

optimized non-systematic LDPC codes have large ACPRs
Prior Work and Summary

- Prior Work
  - Optimized LT codes are not universal [YPN09]
Prior Work and Summary

Prior Work

- Optimized LT codes are not universal [YPN09]
- carefully designed turbo codes have large ACPRs [AFMFR09]
Prior Work and Summary

- Prior Work
  - Optimized LT codes are not universal [YPN09]
  - carefully designed turbo codes have large ACPRs [AFMFR09]
  - systematic LDPC codes perform poorly [MFAFR10]
Prior Work and Summary

- Prior Work
  - Optimized LT codes are not universal [YPN09]
  - carefully designed turbo codes have large ACPRs [AFMFR09]
  - systematic LDPC codes perform poorly [MFAFR10]

- Summary of this work
Prior Work and Summary

- Prior Work
  - Optimized LT codes are not universal [YPN09]
  - carefully designed turbo codes have large ACPRs [AFMFR09]
  - systematic LDPC codes perform poorly [MFAFR10]

- Summary of this work
  - systematic codes $\Rightarrow$ correlated codes $\Rightarrow$ suboptimal
Prior Work and Summary

- Prior Work
  - Optimized LT codes are not universal [YPN09]
  - carefully designed turbo codes have large ACPRs [AFMFR09]
  - systematic LDPC codes perform poorly [MFAFR10]

- Summary of this work
  - systematic codes ⇒ correlated codes ⇒ suboptimal
  - optimized non-systematic LDPC codes have large ACPRs
Tanner Graph & Density Evolution

\[ \begin{align*}
\lambda(x) & \quad \cdots \quad \cdots \cdots \\
\rho(x) & \quad \text{permutation } \pi_1 \\
\end{align*} \]
Tanner Graph & Density Evolution

\[ \rho(x) \]
\[ \lambda(x) \]

**permutation** \( \pi_2 \)

\[ \gamma \]
\[ \lambda(x) \]
\[ \rho(x) \]

**permutation** \( \pi_1 \)
Tanner Graph & Density Evolution

\[ \rho(x) \]

\[ \lambda(x) \]

\[ \text{permutation } \pi_2 \]

\[ \text{permutation } \pi_1 \]

\[ f(\cdot) \]

\[ p \]

\[ \gamma \]
Tanner Graph & Density Evolution

\[ \begin{align*}
\rho(x) & \quad \cdots \quad \pi_2 \quad \cdots \\
\lambda(x) & \quad \cdots \\
\gamma & \quad f(\cdot) \\
\lambda(x) & \quad \cdots \\
\rho(x) & \quad \cdots
\end{align*} \]

BMSC

BMSC

BMSC

BMSC
Tanner Graph & Density Evolution

\[ \rho(x) \]
\[ \lambda(x) \]
\[ \rho(x) \]
\[ \lambda(x) \]

\[ \text{permutation } \pi_2 \]

\[ b_\ell \]
\[ a_\ell \]

\[ \text{permutation } \pi_1 \]

\[ p \]
\[ f(\cdot) \]
\[ \gamma \]

\[ b_{BMSC} \]
\[ a_{BMSC} \]
Tanner Graph & Density Evolution

\[ a_{\ell+1} = \left[ \begin{array}{c} \end{array} \right] \otimes \lambda(\rho(a_\ell)) \]
\[ a_{\ell+1} = \left[ (1 - \gamma)a_{\text{BMSC}} \right] \otimes \lambda(\rho(a_\ell)) \]
Tanner Graph & Density Evolution

$$L(\rho(b_\ell))$$

$$a_{\ell+1} = \left[ \gamma f(L(\rho(b_\ell))) + (1 - \gamma) a_{BMSC} \right] \odot \lambda(\rho(a_\ell))$$
Tanner Graph & Density Evolution

\[
\begin{align*}
a_{\ell+1} &= \left[ \gamma f\left( L(\rho(b_\ell)) \right) + (1 - \gamma) a_{BMSC} \right] \otimes \lambda(\rho(a_\ell)) \\
b_{\ell+1} &= \left[ \gamma f\left( L(\rho(a_\ell)) \right) + (1 - \gamma) b_{BMSC} \right] \otimes \lambda(\rho(b_\ell))
\end{align*}
\]
Tanner Graph & Density Evolution

\[ \Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) - \text{residual error probability} \]
Code Design

- fix the desired rate $R_d$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$

\[ \text{cost function: } A_x = \left\{ \left( \alpha_1; \alpha_2 \right) \mid \alpha_1 \alpha_2 \in C \right\} \]

\[ F(x) = a \sum \left( \alpha_1; \alpha_2 \right)^2 C_1 - C \left( A_x \right) + b \left( R_d - R(x) \right) \]

search using differential evolution
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require *convergence* on $C$

$$\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \forall (\alpha_1, \alpha_2) \in C$$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require convergence on $C$

$$\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \forall (\alpha_1, \alpha_2) \in C$$

- cost function: $(x = [\lambda \ \rho])$

$$A_x = \{(\alpha_1, \alpha_2) | \Gamma_x(\alpha_1, \alpha_2) \leq (\tau, \tau)\}$$

$$F(x) =$$

$$+$$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require convergence on $C$

$$
\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \ \forall \ (\alpha_1, \alpha_2) \in C
$$

- cost function: $(x = [\lambda \ \rho])$

$$
A_x = \{(\alpha_1, \alpha_2) | \Gamma_x(\alpha_1, \alpha_2) \leq (\tau, \tau)\}
$$

$$
\mathcal{F}(x) = \sum_{(\alpha_1, \alpha_2) \in \mathbb{C}} \left(1 - \mathbb{1}_{\{(\alpha_1, \alpha_2) \in A_x\}}\right)
$$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require *convergence* on $C$

$$\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \quad \forall \ (\alpha_1, \alpha_2) \in C$$

- cost function: $(x = [\lambda \ \rho])$

$$A_x = \{ (\alpha_1, \alpha_2) | \Gamma_x(\alpha_1, \alpha_2) \leq (\tau, \tau) \}$$

$$\mathcal{F}(x) = \sum_{(\alpha_1, \alpha_2) \in C} \left( 1 - \mathbb{1}_{(\alpha_1, \alpha_2) \in A_x} \right)$$

$$+ \quad (R_d - R(x))$$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require convergence on $C$

$$\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \forall (\alpha_1, \alpha_2) \in C$$

- cost function: $(x = [\lambda \ \rho])$

$$A_x = \{(\alpha_1, \alpha_2) | \Gamma_x(\alpha_1, \alpha_2) \leq (\tau, \tau)\}$$

$$\mathcal{F}(x) = a \cdot \sum_{(\alpha_1, \alpha_2) \in C} \left(1 - \mathbb{1}_{\{(\alpha_1, \alpha_2) \in A_x\}}\right) + b \cdot (R_d - R(x))$$
Code Design

- fix the desired rate $R_d$
- pick a set of channel conditions $C$
- require *convergence* on $C$

$$\Gamma_{\lambda, \rho}(\alpha_1, \alpha_2) \leq (\tau, \tau) \forall (\alpha_1, \alpha_2) \in C$$

- cost function: $(x = [\lambda \; \rho])$

$$A_x = \{(\alpha_1, \alpha_2) | \Gamma_x(\alpha_1, \alpha_2) \leq (\tau, \tau)\}$$

$$F(x) = a \cdot \sum_{(\alpha_1, \alpha_2) \in C} \left(1 - \mathbb{1}_{(\alpha_1, \alpha_2) \in A_x}\right)$$

$$+ b \cdot (R_d - R(x))$$

- search using differential evolution
Code Design contd.,

- BSC correlated sources
Code Design contd.,

- BSC correlated sources
  - \( \Pr(U_1 = U_2) = p \), \( \Pr(U_1) = \Pr(U_2) = 1/2 \)

\[
f(a) = a \cdot (1 - p) + p \cdot a
\]
Code Design contd.,

- BSC correlated sources
  - \( \text{Pr}(U_1 = U_2) = p, \text{Pr}(U_1) = \text{Pr}(U_2) = 1/2 \)
  - all-zero codeword assumption not valid
Code Design contd.,

- BSC correlated sources
  - \( \Pr(U_1 = U_2) = p, \Pr(U_1) = \Pr(U_2) = 1/2 \)
  - all-zero codeword assumption not valid
  - \( f(a) = a_{BSC(1-p)} \ast a \)

\[ a_{BSC} \]

\[ a \]
• BSC correlated sources
  • $\Pr(U_1 = U_2) = p$, $\Pr(U_1) = \Pr(U_2) = 1/2$
  • all-zero codeword assumption not valid
  • $f(a) = a_{BSC}(1-p) \ast a$
  • transmission through AWGN channels
Code Design contd.,

- **BSC correlated sources**
  - $\text{Pr}(U_1 = U_2) = p$, $\text{Pr}(U_1) = \text{Pr}(U_2) = 1/2$
  - all-zero codeword assumption not valid
  - $f(a) = a_{BSC}(1-p) \ast a$
  - transmission through AWGN channels

- **erasure correlated sources**
Code Design contd.,

- **BSC correlated sources**
  - $\Pr(U_1 = U_2) = p$, $\Pr(U_1) = \Pr(U_2) = 1/2$
  - all-zero codeword assumption not valid
  - $f(a) = a_{BSC(1-p)} \star a$
  - transmission through AWGN channels

- **erasure correlated sources**
  - $Z \sim \text{Ber}(p)$
  - $(U_1, U_2) = \begin{cases} 
  \text{i.i.d. Bernoulli } \frac{1}{2} \text{ r.v.s, if } Z = 0 \\
  \text{same Bernoulli } \frac{1}{2} \text{ r.v. } U, \text{ if } Z = 1
  \end{cases}$
Code Design contd.,

- **BSC correlated sources**
  - $\Pr(U_1 = U_2) = p$, $\Pr(U_1) = \Pr(U_2) = 1/2$
  - All-zero codeword assumption not valid
  - $f(a) = a_{BSC(1-p)} \ast a$
  - Transmission through AWGN channels

- **Erasure correlated sources**
  - $Z \sim \text{Ber}(p)$
  - $(U_1, U_2) = \begin{cases} 
  \text{i.i.d. Bernoulli } \frac{1}{2} \text{ r.v.s, if } Z = 0 \\
  \text{same Bernoulli } \frac{1}{2} \text{ r.v. } U, \text{ if } Z = 1 
  \end{cases}$
  - $f(a) = (1 - p) + pa$
Code Design contd.,

- **BSC correlated sources**
  - $\Pr(U_1 = U_2) = p$, $\Pr(U_1) = \Pr(U_2) = 1/2$
  - all-zero codeword assumption not valid
  - $f(a) = a_{BSC}(1-p) \ast a$
  - transmission through AWGN channels
- **erasure correlated sources**
  - $Z \sim \text{Ber}(p)$
  - \[(U_1, U_2) = \begin{cases} 
    \text{i.i.d. Bernoulli } \frac{1}{2} \text{ r.v.s, if } Z = 0 \\
    \text{same Bernoulli } \frac{1}{2} \text{ r.v. } U \text{, if } Z = 1
  \end{cases}\]
  - $f(a) = (1 - p) + pa$
  - transmission through erasure channels
Results: AWGN ($p = 0.9$)

code rate = 0.282
Results: AWGN ($p = 0.9$)

code rate = 0.282

![Graph showing SNR1 vs. SNR2 with ACPR and SW region shaded areas, with points at (-2.1, -2.1) and (-2.68, -2.68) and a SNR gain of 0.58 dB.]
Results: AWGN ($p = 0.9$)

code rate $= 0.282$

block length $= 10^5$
Results: BEC \( (p = 0.5) \)

code rate = 0.330
Results: BEC ($p = 0.5$)

code rate $= 0.330$
Results: BEC ($p = 0.5$)

code rate = 0.330

block length = $10^5$
Thank You!

