When do Structured Codes Help in Distributed Compression?

Aaron Wagner
Cornell University
The Problem

When do Structured Codes Help in Distributed Compression?

$y_1, \ldots, y_L, x_1, \ldots x_M$: jointly Gaussian scalars
The Problem

\[ y_1, \ldots, y_L, x_1, \ldots x_M: \text{jointly Gaussian scalars} \]

**Quadratic Distortion Constraint:**

\[
\frac{1}{n} \sum_{i=1}^{n} E[(x_\ell(i) - \hat{x}_\ell(i))^2] \leq d_\ell \quad \text{for all } \ell
\]
The Problem

\[ y_1^n, \ldots, y_L^n, x_1^n, \ldots, x_M^n : \text{jointly Gaussian scalars} \]

Rate Region?
The Problem
“Unstructured” Scheme

- Quantize $y_1^n$ in $\Lambda_f$ to $\tilde{y}_1^n$
- Quantize $y_2^n$ in $\Lambda_f$ to $\tilde{y}_2^n$
- Send $\tilde{y}_1^n \mod \Lambda_c$
- Send $\tilde{y}_2^n \mod \Lambda_m$
- Decoder recovers $\tilde{y}_1^n, \tilde{y}_2^n$
- MMSE
“Unstructured” Scheme

- Quantize $y_1^n$ in $\Lambda_f$ to $\tilde{y}_1^n$
- Quantize $y_2^n$ in $\Lambda_f$ to $\tilde{y}_2^n$
- Send $\tilde{y}_1^n$ mod $\Lambda_c$
- Send $\tilde{y}_2^n$ mod $\Lambda_m$
- Decoder recovers $\tilde{y}_1^n, \tilde{y}_2^n$
- MMSE
“Unstructured” Scheme

- Quantize $y_1^n$ in $\Lambda_f$ to $\tilde{y}_1^n$
- Quantize $y_2^n$ in $\Lambda_f$ to $\tilde{y}_2^n$
- Send $\tilde{y}_1^n \mod \Lambda_c$
- Send $\tilde{y}_2^n \mod \Lambda_m$
- Decoder recovers $\tilde{y}_1^n, \tilde{y}_2^n$
- MMSE
“Unstructured” Scheme

- Quantize $y^n_1$ in $\Lambda_f$ to $\tilde{y}^n_1$
- Quantize $y^n_2$ in $\Lambda_f$ to $\tilde{y}^n_2$
- Send $\tilde{y}^n_1 \mod \Lambda_c$
- Send $\tilde{y}^n_2 \mod \Lambda_m$
- Decoder recovers $\tilde{y}^n_1, \tilde{y}^n_2$
- MMSE
“Unstructured” Scheme

<table>
<thead>
<tr>
<th></th>
<th>Var((\tilde{y}_1))</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\tilde{y}_2^n)</td>
</tr>
<tr>
<td></td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td>(\Lambda_c)</td>
<td>(\times)</td>
<td>(\Lambda_m)</td>
<td>(\times)</td>
</tr>
<tr>
<td></td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td></td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\tilde{y}_2^n)</td>
</tr>
<tr>
<td></td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td>(\Lambda_f)</td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
</tbody>
</table>

- Quantize \(y_1^n\) in \(\Lambda_f\) to \(\tilde{y}_1^n\)
- Quantize \(y_2^n\) in \(\Lambda_f\) to \(\tilde{y}_2^n\)
- Send \(\tilde{y}_1^n\) mod \(\Lambda_c\)
- Send \(\tilde{y}_2^n\) mod \(\Lambda_m\)
- Decoder recovers \(\tilde{y}_1^n, \tilde{y}_2^n\)
- MMSE
CEO Problem

[formulated by Viswanathan and Berger ’97]

\[ x^n \xrightarrow{y_1^n} \text{Enc 1} \xrightarrow{nR_1} \hat{x}^n \]
\[ x^n \xrightarrow{y_2^n} \text{Enc 2} \xrightarrow{nR_2} \hat{x}^n \]

**Theorem** (Oohama ’99/’05; Prabhakaran, Tse, and Ramchandran ’04): Unstructured scheme is rate-region optimal.

- Key: How much information must inevitably be transmitted about the observation noise?
CEO Reduction

\[ K_y = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \quad \rho > 0 \]
CEO Reduction

\[ K_y = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \quad \rho > 0 \]

\[ x^n = y_1^n + y_2^n \]

\[ \hat{y}_1^n + y_2^n \]
CEO Reduction

\[ K_y = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \quad \rho > 0 \]
CEO Reduction

- Unstructured scheme is optimal
- Works for any positive linear combination
- Equivalence class of $x$ variables

$$x_1 \equiv x_2 \text{ if } E[x_1 | y_1, y_2] = E[x_2 | y_1, y_2]$$
Unstructured Scheme is Not Always Optimal

\[ K_y = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \quad \rho > 0 \]
Unstructured Scheme is Not Always Optimal

\[ \Lambda_c \]

\[ \tilde{y}_2^n \]

\[ \tilde{y}_1^n \]

\[ \Lambda_m \]

\[ \Lambda_f \]
Unstructured Scheme is Not Always Optimal

\[ \Lambda_f \]

\[ \Lambda_m \]

\[ \Lambda_c \]

\[ y_1^n \]

\[ y_2^n \]
Unstructured Scheme is Not Always Optimal

\[
\begin{array}{cccc}
\times & \times & \times & \times \\
\times & \times & \times & \times \\
\times & \times & \times & \times \\
\times & \times & \times & \times \\
\end{array}
\]

\(\Lambda_c\)

\(\Lambda_m\)

\(\Lambda_f\)

\[\text{Var}(\tilde{y}_1)\]

\[\text{Var}(\tilde{y}_1 - \tilde{y}_2)\]
Unstructured Scheme is Not Always Optimal

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Var(\(\tilde{y}_1\))

\[\Lambda_f\]

\[\Lambda_m\]

Körner and Marton ’79

Krithivasan and Pradhan ’07
Structured vs. Unstructured

\[ R_1 = R_2 = \frac{1}{2} \log \left( \frac{2 \text{var}(y_1 - y_2)}{D} \right) \]
The “Warm Grape Juice Problem” [Wagner ‘11]

- **Issue:** $\tilde{y}_1^n - \tilde{y}_2^n \mod \Lambda_m$ not uniformly distributed over $\Lambda_m$ cell
- **Fix:**
  - View $\tilde{y}_1^n \mod \Lambda_m$ and $\tilde{y}_2^n \mod \Lambda_m$ as supersymbols
  - Apply lossless Körner-Marton trick to them
  - Construction $A + $ Körner-Marton in finite vector spaces
- **Good:** Closes gap
- **Bad:** Performance not computable in infinite dimensions
Structured vs. Unstructured

![Graph showing the sum rate (bits per sample) vs. mean square error (D) for different schemes: Unstructured (SQ), Unstructured (VQ), Improved Scheme (SQ), and KP '07 Scheme (VQ).]
Structured vs. Unstructured

- **Open Problem**: Can we do better than time-sharing between KP and unstructured?
Structured vs. Unstructured

- Extended in different direction by Maddah-Ali and Tse ’10
  - Complete story for two encoders and one reproduction
Many Encoders, One Constraint

\[ 1/n \sum_{i=1}^{n} E[(x_1(i) - \hat{x}_1(i))^2] \leq d \]
Theorem (Oohama ’99/’05; Prabhakaran, Tse, and Ramchandran ’04): Unstructured scheme is rate-region optimal
Beyond CEO

**Theorem** (Tavildar, Viswanath, Wagner ’10): Unstructured scheme achieves the entire rate region if
Beyond CEO

**Theorem** (Tavildar, Viswanath, Wagner ’10): Unstructured scheme achieves the entire rate region if

\[ y_n^1, \ldots, y_n^L, x_1^1 \text{ can be embedded in a Markov random field that is a tree.} \]
Theorem (Tavildar, Viswanath, Wagner ’10): Unstructured scheme achieves the entire rate region if

\[ y^n_1, \ldots, y^n_L, x^n_1 \] can be embedded in a Markov random field that is a tree.
cf. CEO

Improves in two ways:

- Tree can have depth > 1
- May add latent variables to construct the tree
When do Variables Themselves form a Tree?

**Theorem:** (Speed and Kiiveri ’86): A given set of random variables satisfies the requirements of a given graph iff

\[ y_i^n \bigcirc y_j^n \quad \text{implies} \quad K_y^{−1}(i, j) = 0 \]
Embeddability for Three Variables

- Three variables can be embedded in a Markov tree if and only if
Embeddability for Three Variables

- Three variables can be embedded in a Markov tree if and only if

\[ |\rho_{ij}| \geq |\rho_{ik}\rho_{kj}| \quad \text{for all } i, j, \text{ and } k \]

\[ \rho_{12}\rho_{13}\rho_{23} \geq 0 \]
Embeddability for Three Variables

- Three variables can be embedded in a Markov tree if and only if

\[ |\rho_{ij}| \geq |\rho_{ik}\rho_{kj}| \quad \text{for all } i, j, \text{ and } k \]

\[ \rho_{12}\rho_{13}\rho_{23} \geq 0 \]

\[ d_{ij} = -\log |\rho_{ij}| \]
Embeddability for Three Variables

- Three variables can be embedded in a Markov tree if and only if

\[ d_{ij} \leq d_{ik} + d_{kj} \text{ for all } i, j, \text{ and } k \]

\[ \rho_{12} \rho_{13} \rho_{23} \geq 0 \]
Embeddability for Three Variables

- Three variables can be embedded in a Markov tree if and only if

\[ d_{ij} \leq d_{ik} + d_{kj} \quad \text{for all } i, j, \text{ and } k \]

\[ \rho_{12}\rho_{13}\rho_{23} \geq 0 \]
Embeddability for More than Three Variables


**Open Problem:** Given $y_1, \ldots, y_L, x_1$, when does there exist $x_2 \equiv x_1$ such that $y_1, \ldots, y_L, x_2$ can be embedded in a Markov tree?
Two Constraints

\[ \frac{1}{n} \sum_{i=1}^{n} E[(x_1(i) - \hat{x}_1(i))^2] \leq d_1 \]

\[ \frac{1}{n} \sum_{i=1}^{n} E[(x_2(i) - \hat{x}_2(i))^2] \leq d_2 \]
Compressing Neighbors

Proposition (Laourine and Wagner ’11): Unstructured scheme is sum rate optimal if the source can be embedded in a depth-two Markov tree with $x_1$ and $x_2$ as neighbors.
Why Neighbors?

\[(\text{Sum Rate}) = (\text{Enc. } A \text{ Rate}) + (\text{Enc. } B \text{ Rate}) \]
\[+ (\text{Enc. } A \text{ Penalty}) + (\text{Enc. } B \text{ Penalty})\]
For quadratic Gaussian distributed rate-distortion

- Unstructured scheme is sometimes optimal
Conclusion

For quadratic Gaussian distributed rate-distortion

- Unstructured scheme is sometimes optimal
- Codes with algebraic structure sometimes beat it
Conclusion

For quadratic Gaussian distributed rate-distortion

- Unstructured scheme is sometimes optimal
- Codes with algebraic structure sometimes beat it
- Determining which is the case uses algebraic structure of covariance matrices
Conclusion

For quadratic Gaussian distributed rate-distortion

- Unstructured scheme is sometimes optimal
- Codes with algebraic structure sometimes beat it
- Determining which is the case uses algebraic structure of covariance matrices
Embedding Example

\[ K_y = \begin{bmatrix} 
1 & 1/4 & 1/4 \\
1/4 & 1 & 1/4 \\
1/4 & 1/4 & 1 \\
\end{bmatrix} \]
Embedding Example

\[ K_y = \begin{bmatrix} 1 & 1/4 & 1/4 \\ 1/4 & 1 & 1/4 \\ 1/4 & 1/4 & 1 \end{bmatrix} \]

\[ K_y^{-1} = \frac{1}{9} \begin{bmatrix} 10 & -2 & -2 \\ -2 & 10 & -2 \\ -2 & -2 & 10 \end{bmatrix} \]
Embedding Example

\[ K_y = \begin{bmatrix}
1 & 1/4 & 1/4 \\
1/4 & 1 & 1/4 \\
1/4 & 1/4 & 1
\end{bmatrix} \]

\[ K_y^{-1} = \frac{1}{9} \begin{bmatrix}
10 & -2 & -2 \\
-2 & 10 & -2 \\
-2 & -2 & 10
\end{bmatrix} \]
Embedding Example

\[ y_0 \sim \mathcal{N}(0, 1) \]
Embedding Example

\[ y_0 \sim \mathcal{N}(0, 1) \]

\[ y_1 = \frac{1}{2} y_0 + z_1 \]

\[ y_2 = \frac{1}{2} y_0 + z_2 \]

\[ y_3 = \frac{1}{2} y_0 + z_3 \]
Embedding Example

\[ y_0 \sim \mathcal{N}(0, 1) \]

\[ y_1 = \frac{1}{2} y_0 + z_1 \]
\[ y_2 = \frac{1}{2} y_0 + z_2 \]
\[ y_3 = \frac{1}{2} y_0 + z_3 \]
Embedding Example

\[ y_0 \sim \mathcal{N}(0, 1) \]

\[ y_1 = \frac{1}{2}y_0 + z_1 \]

\[ y_2 = \frac{1}{2}y_0 + z_2 \]

\[ y_3 = \frac{1}{2}y_0 + z_3 \]

\[
K_y = \begin{bmatrix}
1 & 1/2 & 1/2 & 1/2 \\
1/2 & 1 & 1/4 & 1/4 \\
1/2 & 1/4 & 1 & 1/4 \\
1/2 & 1/4 & 1/4 & 1
\end{bmatrix}
\]
Embedding Example

\[ y_0 \sim \mathcal{N}(0, 1) \]

\[ y_1 = \frac{1}{2}y_0 + z_1 \]

\[ y_2 = \frac{1}{2}y_0 + z_2 \]

\[ y_3 = \frac{1}{2}y_0 + z_3 \]

\[ K_y = \begin{bmatrix}
1 & 1/2 & 1/2 & 1/2 \\
1/2 & 1 & 1/4 & 1/4 \\
1/2 & 1/4 & 1 & 1/4 \\
1/2 & 1/4 & 1/4 & 1
\end{bmatrix} \]

\[ K_y^{-1} = \frac{2}{3} \begin{bmatrix}
3 & -1 & -1 & -1 \\
-1 & 2 & 0 & 0 \\
-1 & 0 & 2 & 0 \\
-1 & 0 & 0 & 2
\end{bmatrix} \]
Three Neighbors

Conjecture: Unstructured scheme achieves the entire rate region if the source can be embedded a Markov tree with $x_1$, $x_2$, and $x_3$ as neighbors.
One-Help-Two Problem

Suppose \( y_0, y_1, \) and \( y_2 \) can be embedded in a tree.

\[
\frac{1}{n} \sum_{m=1}^{n} E[(y_0(m) - \hat{y}_0(m))^2] \leq d_0
\]

\[
\frac{1}{n} \sum_{m=1}^{n} E[(y_1(m) - \hat{y}_1(m))^2] \leq d_1
\]
One-Help-Two Problem

Suppose $y_0$, $y_1$, and $y_2$ can be embedded in a tree.

$$\frac{1}{n} \sum_{m=1}^{n} E[(y_0(m) - \hat{y}_0(m))^2] \leq d_0$$

$$\frac{1}{n} \sum_{m=1}^{n} E[(y_1(m) - \hat{y}_1(m))^2] \leq d_1$$

$$\frac{1}{n} \sum_{m=1}^{n} E[(x(m) - \hat{x}(m))^2] \leq d$$