Distributed Source Coding for Wireless Sensor Networks

Mark Perillo
March 14, 2007
Overview

- Review of information Theory Principles
- Distributed Source Coding Principles
  - Lossless
  - Lossy
  - The CEO Problem
- Practical Distributed Source Coding Schemes
- Ideas For Work In Distributed Source Coding
Review of Information Theory
Principles
Review of Information Theory Principles

- **H(x)**: Entropy - a measure of the information contained of a random variable

\[ H(X) = \sum_i - p_i \log p_i \]

- For Bernoulli random variable X
  - If \( p = 0.5, q=0.5 \) then \( H(X) = H(p) = 1 \) bit
  - If \( p = 0.1, q=0.9 \) then \( H(X) = H(p) = 0.47 \) bit

- For uniform random variable \( Y \in \{1,2,3,\cdots,N\} \)
  \[ H(Y) = \log N \]
Review of Information Theory Principles

- **H(X|Y)**: Conditional entropy - a measure of the information remaining in X, once Y is known

\[
H(X) = \sum_x \sum_y - p(x, y) \log p(x | y)
\]

- For example
  - Uniform random variable X → \( H(X) = \log N \)
  - But if we know Y and we know that X = Y or X=Y+1 with equal probability, \( H(X|Y) = 1 \) bit
Review of Information Theory Principles

- **H(X,Y):** Joint Entropy – a measure of the total information in X and Y

\[
H(X,Y) = \sum_x \sum_y - p(x, y) \log p(x, y)
\]

- **I(X;Y):** Mutual Information – a measure of the amount of information shared by two random variables X and Y

\[
I(X;Y) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)}
\]
Review of Information Theory Principles

$H(X,Y)$

$H(X)$ $H(Y)$

$H(X|Y)$ $I(X;Y)$ $H(Y|X)$
Review of Information Theory Principles

- Source coding
  - A large block of n copies of i.i.d RV X can be compressed into $nH(X)$ bits
  - Based on the theory of typical sets and the Asymptotic Equipartition Property (AEP)
Asymptotic Equipartition Property

- $X_1, X_2, X_3 \ldots X_n$ are i.i.d. random variables
- It is very likely that

\[
-\frac{1}{n} \log p(X_1, X_2, X_3, \ldots, X_n) \approx H(X)
\]

\[
p(X_1, X_2, X_3, \ldots, X_n) \approx 2^{-nH(X)}
\]

- This is a direct result of the weak law of large numbers
Asymptotic Equipartition Property

- If we have a large sequence of random variables, then it is very likely that the drawn sequence will have joint probability about equal to $2^{-nH(x)}$

$$\begin{align*}
(x_1, x_2, x_3, \ldots, x_n) &\in A_\epsilon^{(n)} \\
2^{-n(H(X)+\epsilon)} &\leq p(x_1, x_2, x_3, \ldots, x_n) \leq 2^{-n(H(X)-\epsilon)}
\end{align*}$$

- There are a total of about $2^{nH(x)}$ of these “typical” sequences in the typical set $A_\epsilon^{(n)}$ where nearly all of the probability is concentrated.
Asymptotic Equipartition Property

- If we do a good job compressing the “typical” sequences, the overall quality of the job we do will be good
  - Even if we do a poor job compressing the “atypical” sequences
- In fact, we can compress a block of $X$ with length $n$ into $nH(X)$ bits
- Similarly, a block of $(X,Y)$ with length $n$ can be compressed into $nH(X,Y)$ bits
Jointly Typical Sequences

\[ A_\varepsilon^{(n)} = \left\{ (x^n, y^n) : \begin{align*} &-\frac{1}{n} \log p(x^n) - H(X) < \varepsilon \\ &-\frac{1}{n} \log p(y^n) - H(Y) < \varepsilon \\ &-\frac{1}{n} \log p(x^n, y^n) - H(X,Y) < \varepsilon \end{align*} \right\} \]
Rate Distortion Theory

- The source coding theorem states that in order to reconstruct a discrete input sequence with no loss of information (i.e., no distortion), we must encode at a rate of $H(X)$
- But what if some distortion is allowable?
- And what if we want to describe a continuous random variable with a discrete alphabet?
- What rates are required then?
Rate Distortion Theory

- The answers to this question comes in the form of the rate distortion function $R(D)$

\[
R(D) = \min_{p(\hat{x}|x): \sum_{(x,\hat{x})} p(x) p(\hat{x}|x) d(x,\hat{x}) \leq D} I(X; \hat{X})
\]

- Typical distortion measures:
  - Hamming distance (discrete sources)
  - Mean squared error (continuous sources)
Rate Distortion Theory

- **Examples**
  - Bernoulli Source with Hamming distortion measure
    \[
    R(D) = \begin{cases} 
    H(p) - H(D) & 0 \leq D \leq p \\
    0 & D > p 
    \end{cases}
    \]
  - Gaussian Source with MSE distortion measure
    \[
    R(D) = \begin{cases} 
    \frac{1}{2} \log \frac{\sigma^2}{D} & 0 \leq D \leq \sigma^2 \\
    0 & D > \sigma^2 
    \end{cases}
    \]
Distributed Source Coding Principles
Distributed Source Coding

Example

- Two temperature sensor with 8-bit ADC
- Entropy $H(X) = H(Y) = 8$ bits
- But $Y = X + N, \quad N = \{0,1\}$ with equal probability
- $H(Y|X) \approx 1$ bit
- Collocated encoder:
  - $R = H(X,Y) = H(X) + H(Y|X) = 8 + 1 = 9$ bits
- Separate encoders with no awareness of each other:
  - $R = H(X) + H(Y) = 8 + 8 = 16$ bits
Lossless Distributed Source Coding

$X$  $Y$

Encoder

$H(X,Y)$

Decoder

$R = H(X,Y)$
Lossless Distributed Source Coding

Encoder

$X$

$H(X)$

Decoder

Encoder

$Y$

$H(Y)$

$R = H(X) + H(Y)$
Lossless Distributed Source Coding

$R = H(X) + H(Y|X)$

Can we do this?

Slepian and Wolf proved that you can indeed
Slepian-Wolf Coding

- Can we encode data sequences separately with no rate increase? YES!!!
- Proof is achieved via analysis of an encoding strategy using “binning”
Slepian-Wolf Coding

- Choose rates $R_x$ and $R_y$ to meet Slepian-Wolf requirements
  \[
  R_x = H(X | Y) \\
  R_y = H(Y | X) \\
  R_x + R_y = H(Y, X)
  \]

- Assign every sequence $x^n$ to a random bin on \{1,2,... $2^{nR_x}$\} and every $y^n$ to a random bin on \{1,2,... $2^{nR_y}$\}

- Transmit the bin indices to the decoder

- Look for jointly typical sets in $(x^n, y^n)$ that share indices
Lossy Distributed Source Coding

\[ x^n = \{0, 1, 0, \ldots, 0, 0\} \]
\[ x^n = \{0, 1, 0, \ldots, 0, 1\} \]
\[ x^n = \{0, 1, 0, \ldots, 1, 0\} \]
\[ x^n = \{0, 1, 0, \ldots, 1, 1\} \]

\[ 2^{nR_x} \text{ bins} \]
Lossy Distributed Source Coding

\[ 2^{nH(X,Y)} \] jointly typical sequences

\[ x^n = \{0,1,0,\ldots,1,0\} \]
\[ y^n = \{0,0,0,\ldots,1,0\} \]

Will result in decoding errors
Slepian-Wolf Coding

- When do we get decoding errors?
  - When the X and Y sequences are not jointly typical
    - Rare from Joint AEP
  - When more than one X in the same bin that is jointly typical with Y (or vice versa)
    - Rare if Rx > H(X|Y) (and Ry > H(Y|X))
  - When more than one jointly typical (X,Y) sequence in a single product bin (and vice versa)
    - Rare if Rx + Ry > H(X,Y)
Lossy Distributed Source Coding

\[ R_x \]

\[ R_y \]

\[ H(X) \]

\[ H(X|Y) \]

\[ H(Y|X) \]

\[ H(Y) \]

- Separate Encoders
Slepian-Wolf Coding

- With two variables, we can achieve considerable savings in amount of traffic required.
- In general, for $n$ sensors, the Slepian-Wolf requirement is as follows:

$$R(S) > H(X(S) | X(S^c)) \quad \forall S \subseteq \{1, 2, ..., m\}$$

$$R(S) = \sum_{i \in S} R_i$$

$$X(S) = \{X_j : j \in S\}$$
Source Coding w/ Side Information

- Look at a special case when ONLY X needs to be recovered at the decoder

\[ X \quad \xrightarrow{\text{Encoder}} \quad R_x \quad \xrightarrow{\text{Decoder}} \quad \hat{X} \]

\[ Y \quad \xrightarrow{\text{Encoder}} \quad R_y \]
Source Coding w/ Side Information

- For $U$ such that
  \[ p(x, y, u) = p(x, y)p(u \mid y) \]
  \[ X \rightarrow Y \rightarrow U \]

- Then we can send at rates
  \[ R_x \geq H(X \mid U) \]
  \[ R_y \geq I(Y; U) \]
Source Coding w/ Side Information

- One extreme: $U = Y$

\[ R_x \geq H(X \mid U) = H(X \mid Y) \]
\[ R_y \geq I(Y; U) = I(Y; Y) = H(Y) \]

- Another extreme: $U$ is uncorrelated with $X, Y$

\[ R_x \geq H(X \mid U) = H(X) \]
\[ R_y \geq I(Y; U) = 0 \]

- The results from Wyner basically say that we can operate between these two extremes
If we want to reproduce $X$ with some distortion $D$, what rate must we send at, given that we have access to side information $Y$?

\[ E[d(X, \hat{X})] < D \]
R-D w/ Side Information

- If $X, Y$ are uncorrelated, this is just the rate distortion function $R(D)$
- Since side information can only help,
  
  \[ R_Y^*(D) \leq R(D) \]

- If no distortion is allowed, then
  
  \[ R_Y^*(0) = H(X \mid Y) \]

  as Slepian and Wolf showed
R-D w/ Side Information

- In general,

\[ R^*_Y(D) = \min_{p(w|x)} \min_f (I(X;W) - I(Y;W)) \]

\[ \sum \sum \sum p(x, y) p(w | x) d(x, f(y, w)) \leq D \]

- \( f \) is the reconstruction function

- \( w \) is the encoded version of \( x \)
R-D w/ Side Information

- Remember: in lossless (S-W coding), we pay no penalty for separation of X and Y
- Unfortunately, this is not the case in lossy source coding
- In general, \( R_Y^*(D) \geq R_{X|Y}(D) \)
  \[ R_{X|Y}(D) : \text{rate required when X's encoder has access to Y} \]
- However, equality is achieved when (X,Y) are jointly Gaussian
Distributed Data Compression

- The rate-distortion region for this general problem is unknown

![Diagram of distributed data compression system]
The CEO Problem

Source signal, $x$

Sensor node

Finite rate communication link

Root node
The CEO Problem

\[ X \sim p(X) \]

\[ W(y_1 | x) \rightarrow \text{Encoder} \]
\[ W(y_2 | x) \rightarrow \text{Encoder} \]
\[ W(y_L | x) \rightarrow \text{Encoder} \]

\[ \text{Decoder} \rightarrow \hat{X} \]
The CEO Problem

- If the CEO’s L agents were able to convene, they could smooth the data (i.e., average out the noise) to obtain a true value of X and then send data at a rate R(D) required to keep distortion below D.

- But what happens if they cannot convene?
  - Assume a sum rate requirement $\sum_i R_i \leq R$.
The CEO Problem

- Berger et al originally found the limits of this problem for a Hamming distance measure.

- For $R > H(X)$, $P_e(R)$ does not go to 0.

- In fact, $P_e(R) = 2^{-\alpha(p,W)R}$ as $R$ gets large, where $\alpha(p,W)$ is a constant for a given source distribution and joint source/observation distribution.

- In other words, there is always a penalty for not being able to convene.
The Quadratic Gaussian CEO Problem
The Quadratic Gaussian CEO Problem

- Viswanathan and Berger et al found the limits of this problem as well
- As R and L get very large, what happens to D as a function of sum rate R
The Quadratic Gaussian CEO Problem

- Result

\[ \beta(\sigma_X^2, \sigma_N^2) = \lim_{R \to \infty} \lim_{L \to \infty} R \frac{D(R, L)}{\sigma_X^2} \]

\[ 0 < \inf_{Q(u|y)} \frac{I(Y; U \mid X)}{\sigma_X^2 E \left[ - \frac{\partial^2}{\partial X^2} \log \tilde{Q}(U \mid X) \right]} \leq \beta(\sigma_X^2, \sigma_N^2) \leq \frac{\sigma_N^2}{2\sigma_X^2} \]

- Oohama proved that for Gaussian X,N, these bounds are tight
The Quadratic Gaussian CEO Problem

- This means that distortion decays as \[ D = \frac{\sigma^2}{2R} \]

- Compare with the case when agents can all convene

- Agents can smooth out the data and get rid of noise \[ D = \sigma_X^2 2^{-2R} \]

- So again, a penalty is paid for not being able to convene
The Quadratic Gaussian CEO Problem

• How do we achieve this? \( D = \frac{\sigma^2}{2R} \)
Practical Distributed Source Coding Schemes
Draper et al’s Work

- Side Information aware limit:
  - For a given rate constraint $R$

$$d \geq \min_{f, u \in \Pi} E[D(x, f(y_D, u))]$$

$$u \rightarrow y_E \rightarrow x, y_D, y_N$$

$$R > I(y_E; u) - I(y_D; u)$$
Draper et al’s Work

- How to do this:
  - Construct a code of \( \sim 2^{nI(y_e;u)} \) typical codewords
  - Bin each of these into one of \( 2^{nR} \) bins
    - \( \sim 2^{n(I(y_e;u) - R)} \) codewords per bin
  - Block encode and transmit codeword’s bin (coset index)
  - Decoder choose codeword in coset that is jointly typical with its observation
Draper et al’s Work

- Side Information aware limit:
  - For a given rate constraint $R_{\text{cut}}$

\[
d \geq \min_{f,u \in \Pi} E \left[ D(x, f(y^{A^c}, u)) \right]
\]

\[
u \rightarrow y^A \rightarrow x, y^{A^c}
\]

\[
R_{\text{cut}} > I(y^A; u) - I(y^{A^c}; u)
\]

- One of these constraints exists for all possible cuts
Draper et al’s Work

- Quadratic Gaussian Case:

\[
R(d) = \frac{1}{2} \log \left( \frac{\sigma^2_{x|y_D} - \sigma^2_{x|y_E,y_D}}{d - \sigma^2_{x|y_E,y_D}} \right)
\]

\[
d(R) = \sigma^2_{x|y_E,y_D} + \left( \sigma^2_{x|y_D} - \sigma^2_{x|y_E,y_D} \right) 2^{-2R}
\]

\[
\sigma^2_{x|y_D} : \text{Minimum distortion given decoder observation}
\]

\[
\sigma^2_{x|y_D,y_E} : \text{Minimum distortion given encoder, decoder observation}
\]
Draper et al’s Work

- **Serial Network**

\[
d_l \geq \frac{N_l d_{l-1}}{N_l + d_{l-1}} + \sigma^2_{x|y_l} \left( 1 - \frac{d_{l-1}}{\sigma^2_x} \right) \left( 1 + \frac{d_{l-1}}{N_l} \right)^{-2R_l-1}
\]

\[
d_1 = \sigma^2_{x|y_1}
\]
Draper et al’s Work

- Parallel Network

\[ d_l \geq \frac{N_l d_{l-1}}{N_l + d_{l-1}} + \frac{d_{l-1}^2}{N_l + d_{l-1}} 2^{-2R} \]

\[ d_0 = \sigma_x^2 \]
Draper et al’s Work

- We can apply the serial and parallel results to get the achievable distortion of a general sensor network tree
Draper et al’s Work

- Serial Network
- Source variance = 4
- Noise variance = 4/3
- R = 2.5 bits
Source Coding Using Syndromes

- Originally introduced by Wyner in 1974
- Used by many in the literature recently, notably DISCUS by Pradhan et al.
- Exploits duality between Slepian-Wolf coding and channel coding
DISCUS Example

- $X = \{0,1\}^n$ and $Y = \{0,1\}^n$, $n = 3$
- $X$, $Y$’s correlation is such that they differ in at most 1 bit
- Thus, given $Y$, $X$ can take 4 values:
  - Same or differ in the $1^{st}$, $2^{nd}$, or $3^{rd}$ bit
- $H(X|Y) = 2$
  - If we send all of $Y$, this is the S-W limit
DISCUS Example

- Divide codeword space into 4 cosets
  - 0: {000, 111}, 1: {001, 110}, 2: {010, 101}, 3: {011, 100}
  - Send only the coset index (4 cosets = 2 bits)
  - This meets the S-W bound

- Given Y, only one of the members of the coset can have a Hamming distance of 0 or 1

- Coset index is just the syndrome of x using the parity-check matrix \( H \)

\[
H = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}
\]

\[
s = Hx
\]
DISCUS Example

\[ X = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \]
\[ Y = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \]
\[ s = Hx \]

\[
\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]
i.e., coset index is 2

- Decoder knows that choices for X are [010] and [101]
- Since Y is [110], it has to be [010] since [101] has a Hamming distance of 2
Source Coding Using Syndromes

- Lets say we have correlated $X$ and $Y$
- $X = Y + U$
  - $U$ is a Bernoulli RV with $p < 0.5$
- $Y$ sends full information
  - $R_Y = H(Y)$
- $X$ sends partial information, hopefully at the Slepian-Wolf limit
Source Coding Using Syndromes

- Lets generate a parity check matrix \( H (m \times n) \)

- \( X \)'s encoder generates \( n \) samples and then calculates the syndrome \( s_X = HX \)
  - No guarantee that \( X \) is a valid codeword (in the null space of \( X \)) so \( s_X \) can take any value

- \( X \)'s encoder sends \( s_X \) to the decoder
Source Coding Using Syndromes

- Decoder generates syndrome of \( Y \)
  - \( s_Y = HY \)
- Decoder calculates \( s_X+s_Y \) \( s_{X+Y} = s_U \)
- If \( X \) and \( Y \) are highly correlated, we can expect that there aren’t too many differences
  - i.e., \( U \) should be mostly zeros
- Decoder can now approximate \( U = f(s_U) \) and therefore, approximate \( X \)
Source Coding Using Syndromes

- Alternative way to do this (from DISCUS paper)
  - $s = Hx$
  - $y' = y + [0|s]$
  - Find $x'$, the closest code to $y'$ in the coset with $s = [00...0]$
    - i.e., the closest valid codeword in the null space of $H$
  - $x = x' + [0|s]$
Source Coding Using Syndromes

- In general, if we have a \((n,k,2t+1)\) code,
  - \(R = n - k\)
  - S-W bound is the log of number of possible outcomes for \(X\) given \(Y\), given each is equally likely

\[
R_X^{SW} \log \sum_{i=0}^{t} \binom{n}{i}
\]

- In previous example, \(n=3, k=1, t=1\)
  - \(R_X^{SW} = \log(1+3) = 2\), \(R = n-k = 2\)
Source Coding Using Syndromes

- A \((n,k,2t+1)\) code can be used if the differences between \(X\) and \(Y\) are at most \(t\)

- Let's look at an \(X,Y\) correlation structure that can be modeled as a BSC with crossover probability \(p\)

- Probability of a correct decode is the probability that the number of crossovers \(\leq t\)

\[
P = \sum_{i=0}^{t} \binom{n}{i} (1-p)^{n-i} p^i
\]
The benefit of using good channel codes should be apparent now.

There has been a lot of work recently applying state-of-the-art codes like LDPC and Turbo codes to the S-W problem.

Also, some have used convolutional codes in similar ways.
DISCUS with Continuous Sources

Fictitious channel $P(Y|W)$ exists
$W$ sends to $Y$ at a rate of $I(W;Y)$
This means fewer bits actually need to be sent
Required rate is now $H(W|Y) = H(W) - I(W;Y)$
DISCUS with Continuous Sources

- Lets say that we set the fictitious rate at 1 bit
- This means we need to send two bits
- Lets make our cosets \{1,5\}, \{2,6\}, \{3,7\}, \{4,8\}
- Sample X and then send the coset index
DISCUS with Continuous Sources

- $X$ quantizes to value of 6
- $X$ send coset index 1 $\rightarrow \{2, 6\}$
- Decoder knows $X$ has been quantized to 2 or 6
- Remember that the fictitious channel sent the other bit
- This means that due to knowledge of $Y$ and the correlation structure, the decoder can disambiguate between the 2 possible values (1 bit)
Separate Encoders
Time-Sharing

- Most typical distributed coding schemes operate on the corners of the achievability region
- But you can operate anywhere on the curve
- Easiest way is via time-sharing
  - i.e., X sends full data, Y sends partial data for some portion of time, and then they reverse roles for some portion of time
Critescu et al’s Work

- Compare approaches for correlated data gathering in tree networks
  - S-W model
    - Coding complex, Transmission Optimization simple
  - Joint Entropy Coding model
    - No S-W, encoding occurs after others’ data is explicitly known
    - Coding simple, Transmission Optimization complex

- How does the choice affect rate allocation?
Critescu et al’s Work

- Main result:
  - S-W Coding – allocate high rates to nodes close
    - Since they have to route over fewer hops
    - Liu et al in Mobicom 2006 paper came up with similar results
  - Joint Entropy Coding – allocate high rates to nodes furthest nodes and lower rates to nearby nodes
    - Since they can use others as side information
Critescu et al’s Work

- **SW scheme**
  - Optimize ST and then use LP for rate allocation
  - LP solution has closed form
    \[
    \begin{align*}
    R_1 &= H(X_1) \\
    R_2 &= H(X_2 | X_1) \\
    &\vdots \\
    R_N &= H(X_N | X_{N-1}, \ldots, X_2, X_1)
    \end{align*}
    \]

- **Approximation:**
  - Each node find nodes in neighborhood that are closer on the SPT
  - Transmit at a rate
    \[
    R_i = H(X_i | C_i)
    \]
Critescu et al’s Work

- Clustered SW scheme
  - Since S-W for many nodes is very complex, we can do it in clusters
  - How do we choose clusters such that
    \[
    \{I_i^*\} = \arg \min_{C,\{I_i\}_{i=1}^C} \left( \prod_{i=1}^C \det K_{I_i} \right)
    \]
  - NP-complete problem for all but a few degenerate cases
Critescu et al’s Work

- Joint Entropy Coding scheme
  - Problem is NP-complete
  - Authors provide SPT-based heuristics
Critescu et al’s Work
Sensor Correlation

- A lot of the solutions for limits of distributed source coding assume the specific cases of
  - Binary sources
    - Correlation modeled as a BSC
    - Distortion measured as Hamming distance
  - Jointly Gaussian sources
    - Distortion measured as MSE
Sensor Correlation

- How should we model correlation of real world sensor data?
  - We could use a training phase where the correlation is learned over time
  - We can assume that nearby sensors have higher correlation
    - True in many real-world applications
Sensor Correlation

• Pattem et al’s approach
  – If X and Y are separated by a distance $d(X,Y)$

$$H(X_2 | X_1) = \left( 1 - \frac{1}{d(X_1, X_2) + 1} \right) H(X_2)$$

  – Fit this data to empirical rainfall data
Sensor Correlation

- Additional sensors add and additional
  \[\left(1 - \frac{1}{\frac{d(X_1, X_2)}{c} + 1}\right)H(X_2)\]
  bits of information

- If all sensors equally spaced by d, total information is about
  \[H(X) = H(X_1) + (n - 1)\left(1 - \frac{1}{\frac{d}{c} + 1}\right)H(X_1)\]
Sensor Correlation

- Critescu et al assumes a Gaussian Markov Field

\[
f(X) = \frac{1}{\sqrt{2\pi \det(K)}} e^{-\frac{1}{2}(X-\mu)^T K^{-1} (X-\mu)}
\]

\[
K_{ij} = \sigma_{ij}^2 = \sigma^2 e^{-cd^2}
\]

\[
H(X) = \frac{1}{2} \log(2\pi e)^k \det(K)
\]

\[
H(Y | Y^C) = \frac{1}{2} \log \left( (2\pi e)^{N-\left|Y^C\right|} \frac{\det K}{\det K_{Y^C}} \right)
\]
Sensor Correlation

- Liu et al use 3 models in their analysis
  - Hard Continuity Field  \(|X_1 - X_2| \leq d\)
  - In general, you could use  \(|X_1 - X_2| \leq f(d)\)
  - Linear Covariance Continuity Field  \(E[(X_1 - X_2)^2] \leq d^2\)
  - In general, you could use  \(E[(X_1 - X_2)^2] \leq f(d)\)
  - Gaussian Markov Field

\[ \sigma_{1,2} = \sigma^2 e^{-cd} \]
Ideas For Work In Distributed Source Coding
Ideas For Work In Distributed Source Coding

- There has a lot of work in this field, as applied to WSN, especially in the last 5 or so years.
- Some research has looked at minimizing cost of gathering data in WSN.
  - What about maximizing network lifetime, a la DAPR, MiLAN, etc?
  - What about jointly optimizing transmission ranges with network topology and rate allocation?
Ideas For Work In Distributed Source Coding

- When calculating cost of data gathering, most look at fundamental limits, assuming long block lengths, no overhead
  - What happens when latency is important and we cannot encode long blocks?
    - DISCUS example shows that significant gains can still be made
  - What happens when packet overhead can’t be ignored?
    - e.g., the cost for sending data at 1 bit/sample isn’t much different than sending at 10 bits per sample?
References
References


References


References


References


