# Distributed Lossy Computing

Abbas El Gamal

Stanford University

Based on work with Han-I Su, P. Cuff, Young-Han Kim

Partially supported by DARPA ITMANET

ITA 2010

# Introduction

- Performance of distributed information processing systems often limited by communication

  Digital VLSI

  Multi-processors

  Data centers

  Peer-peer networks

  Sensor networks

  Networked mobile agents

# Introduction

- Performance of distributed information processing systems often limited by communication

  Digital VLSI

  Multi-processors

  Data centers

  Peer-peer networks

  Sensor networks

  Networked mobile agents

- Purpose of communication is to make decision, compute function, coordinate action based on distributed data

# Introduction

- Performance of distributed information processing systems often limited by communication

  Digital VLSI

  Multi-processors

  Data centers

  Peer-peer networks

  Sensor networks

  Networked mobile agents

- Purpose of communication is to make decision, compute function, coordinate action based on distributed data

How much communication is needed to perform such a task?

# Problem Formulations

- Problem formulated and studied in several fields:
  - Computer science: Communication complexity; gossip
  - Information theory: Coding for computing; $\mu$-sum problem
  - Control: Distributed consensus

# Problem Formulations

- Problem formulated and studied in several fields:
  - Computer science: Communication complexity; gossip
  - Information theory: Coding for computing; $\mu$-sum problem
  - Control: Distributed consensus

- Formulations differ in:
  - Data model (discrete, continuous; deterministic, random)
  - Type of coding–computing (scalar, block)
  - Estimation criterion (error-free; lossless; lossy)
  - Metric for communication cost (bits; rate; rounds)

# This Talk: Lossy Distributed Computing

- Problem formulated and studied in several fields:
  - Computer science: Communication complexity; gossip
  - Information theory: Coding for computing; $\mu$-sum problem
  - Control: Distributed consensus
- Formulations differ in:
  - Data model (discrete, continuous; deterministic, random)
  - Type of coding–computing (scalar, block)
  - Estimation criterion (error-free; lossless; lossy)
  - Metric for communication cost (bits; rate; rounds)

# This Talk: Lossy Distributed Computing

- Problem formulated and studied in several fields:
  - Computer science: Communication complexity; gossip
  - Information theory: Coding for computing; $\mu$-sum problem
  - Control: Distributed consensus

- Formulations differ in:
  - Data model (discrete, continuous; deterministic, random)
  - Type of coding–computing (scalar, block)
  - Estimation criterion (error-free; lossless; lossy)
  - Metric for communication cost (bits; rate; rounds)

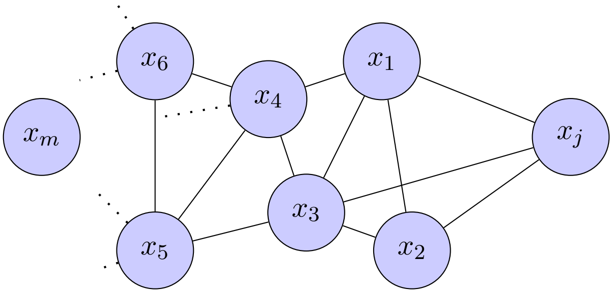
- Advantages of IT formulaion:
  - Many real-world sources have large (continuous) alphabets
  - Limits that hold "in general"
  - Asymptotic approach simplifies analysis

# Distributed Consensus

Distributed coordination, information aggregation in distributed systems

- Flocking and schooling behaviors in nature [Vicsek et al., 95]
- Spread of rumors, epidemics [Demers et al., 87]
- Coordination of autonomous vehicles [Jadbabaie et al., 03]
- Load balancing in parallel computers
- Finding file sizes in peer-to-peer networks [Bawa et al., 03]
- Information aggregation in sensor networks [Kempe et al., 03]

# Distributed Averaging [Olfati-Saber et al., 03]




- Undirected graph $(\mathcal{M}, \mathcal{E})$ with $m$ nodes
- Node $j$ observes real-valued scalar $x_j$
- Each node wishes to estimate the average $s = (1/m)\sum_{j=1}^{m} x_j$ to some prescribed MSE

# Averaging Protocols

- Nodes communicate and perform local computing in *rounds*
  - For example, a node-pair is selected in each round

# Averaging Protocols

- Nodes communicate and perform local computing in *rounds*
  - For example, a node-pair is selected in each round

- Sequence of communication rounds and operations performed in each round determined by an *averaging protocol*

# Averaging Protocols

- Nodes communicate and perform local computing in *rounds*
  - For example, a node-pair is selected in each round

- Sequence of communication rounds and operations performed in each round determined by an *averaging protocol*

- In many applications, communication is asynchronous, subject to link and node failures and topology changes
  - Distributed protocols: Do *not* depend on node identities
    Gossip protocols: Random node subset selections
    [Hedetniemi et al., 88]

# Averaging Protocols

- Nodes communicate and perform local computing in *rounds*
  - For example, a node-pair is selected in each round

- Sequence of communication rounds and operations performed in each round determined by an *averaging protocol*

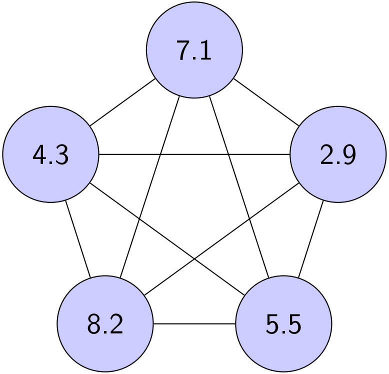
- In many applications, communication is asynchronous, subject to link and node failures and topology changes
  - Distributed protocols: Do *not* depend on node identities
    Gossip protocols: Random node subset selections
    [Hedetniemi et al., 88]

Do the estimates converge to average? How many rounds are needed to achieve prescribed MSE?

# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

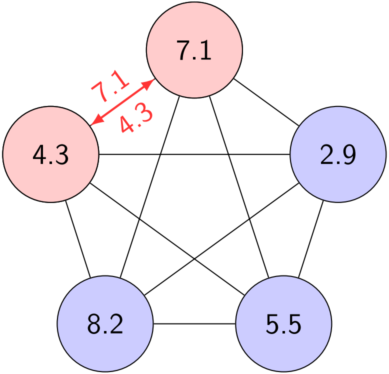
$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$
$$s_j(t+1) = \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t)$$
$$= s_k(t+1)$$



# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

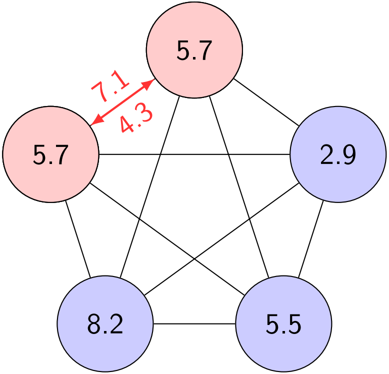
$$
\begin{aligned}
s_j(0) &= x_j,\ j = 1, 2, \ldots, m \\
s_j(t+1) &= \frac{1}{2} s_j(t) + \frac{1}{2} s_k(t) \\
&= s_k(t+1)
\end{aligned}
$$



# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$
\begin{aligned}
s_j(0) &= x_j,\ j = 1, 2, \ldots, m \\
s_j(t+1) &= \frac{1}{2} s_j(t) + \frac{1}{2} s_k(t) \\
&= s_k(t+1)
\end{aligned}
$$



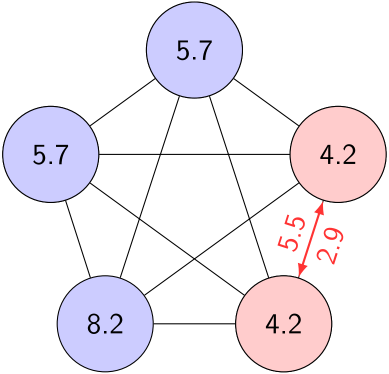
# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$

$$s_j(t+1) = \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t)$$

$$= s_k(t+1)$$

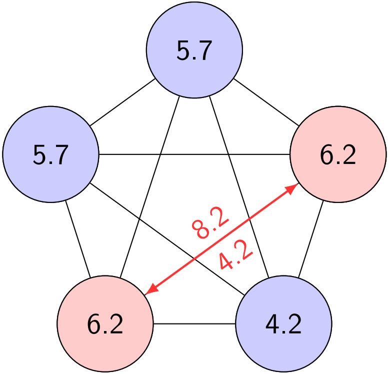


# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$

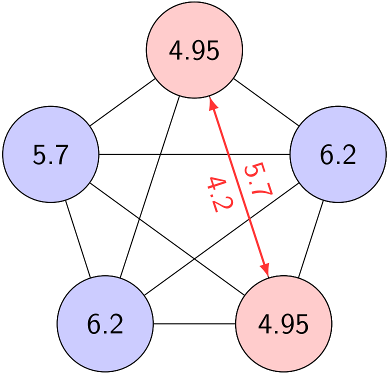
$$s_j(t+1) = \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t) = s_k(t+1)$$



# Example: Gossip Protocol

- In each round, a node-pair $(j,k)$ is randomly selected
- Nodes exchange values and average them

$$\begin{aligned} s_j(0) &= x_j,\ j = 1,2,\ldots,m \\ s_j(t+1) &= \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t) \\ &= s_k(t+1) \end{aligned}$$



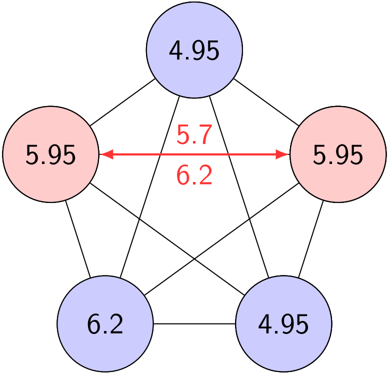
# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$

$$s_j(t+1) = \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t)$$

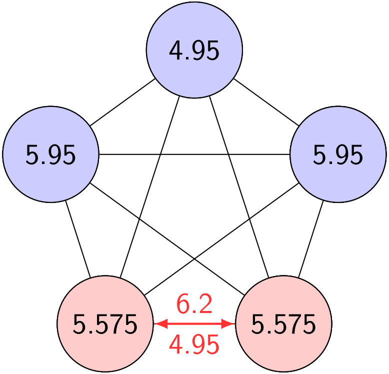
$$= s_k(t+1)$$



# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$
$$s_j(t+1) = \frac{1}{2} s_j(t) + \frac{1}{2} s_k(t)$$
$$= s_k(t+1)$$

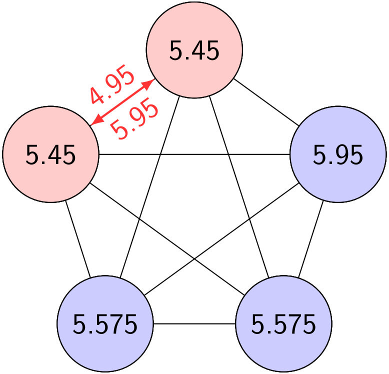


# Example: Gossip Protocol

- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$

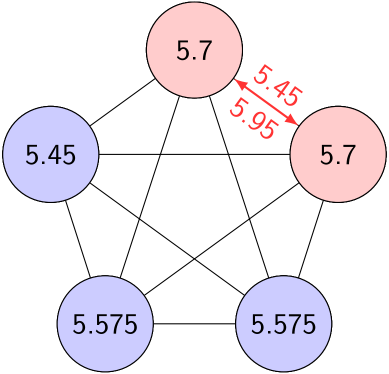
$$s_j(t+1) = \frac{1}{2} s_j(t) + \frac{1}{2} s_k(t) = s_k(t+1)$$



# Example: Gossip Protocol

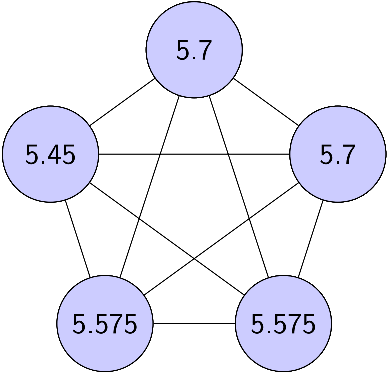
- In each round, a node-pair $(j, k)$ is randomly selected
- Nodes exchange values and average them

$$s_j(0) = x_j,\ j = 1, 2, \ldots, m$$
$$s_j(t+1) = \frac{1}{2}s_j(t) + \frac{1}{2}s_k(t)$$
$$= s_k(t+1)$$



# Example: Gossip Protocol

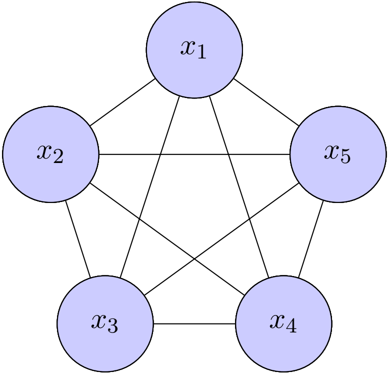
- MSE $= 0.00875$
- Estimates converge to the average
- Bounds on the number of rounds as function of MSE [Boyd et al., 05]



# Example: Gossip Protocol

Issues:

- Communicating/computing with infinite precision not realistic
- Number of rounds not good measure of communication cost



# Quantized Distributed Averaging

- Control: Scalar quantization
  - A. Nedic, A. Olshevsky, A. Ozdaglar, and J. Tsitsiklis, "On distributed averaging algorithms and quantization effects," 2007
  - A. Kashyap, T. Basar, and R. Srikant, "Quantized consensus," *Automatica*, 2007

# Quantized Distributed Averaging

- Control: Scalar quantization
  - A. Nedic, A. Olshevsky, A. Ozdaglar, and J. Tsitsiklis, "On distributed averaging algorithms and quantization effects," 2007
  - A. Kashyap, T. Basar, and R. Srikant, "Quantized consensus," *Automatica*, 2007

- Signal Processing: Model quantization noise as additive noise
  - L. Xiao, S. Boyd, S.J. Kim, "Distributed average consensus with least-mean-square deviation," *J. Parallel Distrib. Comput.*, 2007
  - M. Yildiz, A. Scaglione, "Coding with side information for rate constrained consensus," *IEEE Trans. on Sig. Proc.*, 2008

# Quantized Distributed Averaging

- Control: Scalar quantization
  - A. Nedic, A. Olshevsky, A. Ozdaglar, and J. Tsitsiklis, "On distributed averaging algorithms and quantization effects," 2007
  - A. Kashyap, T. Basar, and R. Srikant, "Quantized consensus," *Automatica*, 2007
- Signal Processing: Model quantization noise as additive noise
  - L. Xiao, S. Boyd, S.J. Kim, "Distributed average consensus with least-mean-square deviation," *J. Parallel Distrib. Comput.*, 2007
  - M. Yildiz, A. Scaglione, "Coding with side information for rate constrained consensus," *IEEE Trans. on Sig. Proc.*, 2008
- Information Theory: Treat quantization noise indirectly as link capacity constraint
  - O. Ayaso, D. Shah, M. Dahleh, "Information theoretic bounds on distributed computation," preprint, 2008

# Distributed Lossy Averaging [Su, EG 09]

- Graph with $m$ nodes. Node $j$ observes source $X_j$
- Assume $(X_1, \ldots, X_m)$ jointly Gaussian
- Each node wishes to estimate $S^n = (1/m)\sum_{j=1}^{m} X_j^n$

# Distributed Lossy Averaging [Su, EG 09]

- Graph with $m$ nodes. Node $j$ observes source $X_j$
- Assume $(X_1, \ldots, X_m)$ jointly Gaussian
- Each node wishes to estimate $S^n = (1/m) \sum_{j=1}^{m} X_j^n$
- Averaging protocol:
  - Number of communication rounds $T$
  - Sequence of node pairs selected (deterministic or random)
  - Block code (two-way) used by each selected pair in each round

# Distributed Lossy Averaging [Su, EG 09]

- Graph with $m$ nodes. Node $j$ observes source $X_j$
- Assume $(X_1, \ldots, X_m)$ jointly Gaussian
- Each node wishes to estimate $S^n = (1/m)\sum_{j=1}^m X_j^n$
- Averaging protocol:
  - Number of communication rounds $T$
  - Sequence of node pairs selected (deterministic or random)
  - Block code (two-way) used by each selected pair in each round
- Let $R$ be sum rate over the $T$ rounds
- Let $S_j^n(T)$ be estimate of node $j$ at end of round $T$
- Per-letter MSE distortion:

$$D_j^{(n)} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{E}[(S_i - S_{ji}(T))^2]$$

# Distributed Lossy Averaging

- For fixed $T$ and node-pair selections:
  - Rate distortion pair $(R, D)$ achievable if there exists a sequence of codes with rate $R$ such that

$$\limsup_{n\to\infty} \frac{1}{m} \sum_{j=1}^{m} D_j^{(n)} \leq D$$

  - $R(D) = \inf\{R : (R, D) \text{ is achievable}\}$

# Distributed Lossy Averaging

- For fixed $T$ and node-pair selections:
  - Rate distortion pair $(R, D)$ achievable if there exists a sequence of codes with rate $R$ such that

$$\limsup_{n\to\infty} \frac{1}{m} \sum_{j=1}^{m} D_j^{(n)} \le D$$

  - $R(D) = \inf\{R : (R, D) \text{ is achievable}\}$
- The network rate–distortion function:

$$R^*(D) = \inf\{R(D) : \text{ all node-pair selection sequences and } T\}$$

# Distributed Lossy Averaging

- For fixed $T$ and node-pair selections:
  - Rate distortion pair $(R, D)$ achievable if there exists a sequence of codes with rate $R$ such that

$$\limsup_{n\to\infty} \frac{1}{m} \sum_{j=1}^{m} D_j^{(n)} \le D$$

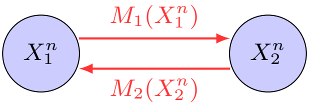
  - $R(D) = \inf\{R : (R, D) \text{ is achievable}\}$
- The network rate–distortion function:

$$R^*(D) = \inf\{R(D) : \text{ all node-pair selection sequences and } T\}$$

- $R^*(D)$ is not known in general

# $R^*(D)$ for 2-Node Network

- 2-nodes; $(X_1, X_2)$ 2-WGN$(1, \rho)$; arbitrary number of rounds
- Nodes wish to estimate average $S^n = (X_1^n + X_2^n)/2$ with MSE distortion $D$



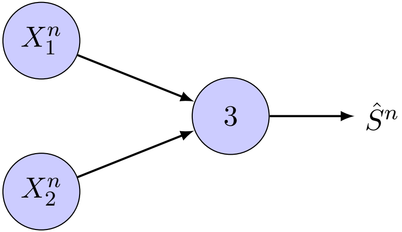

**Theorem (Su, EG 09)**

$$R^*(D) = \log\left(\frac{1-\rho^2}{4D}\right)$$

- Achieved using two independent Wyner-Ziv coding rounds

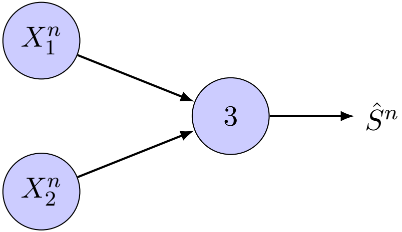
# 2-Source Distributed Lossy Averaging

- Three nodes; $(X_1, X_2)$ 2-WGN(1,$\rho$)
- Communication: $1 \to 3$, $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$



# 2-Source Distributed Lossy Averaging

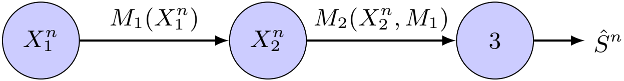
- Three nodes; $(X_1, X_2)$ 2-WGN(1,$\rho$)
- Communication: $1 \to 3$, $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$




- Problem recently solved ($\mu$-sum) by Tse, Ramchandran 04; Oohama 05; Wagner,Tavildar,Viswanath 08

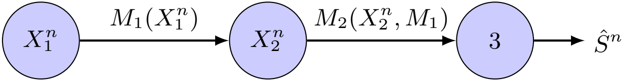
# 3-Node Cascade Network [Cuff, Su, EG 09]

- Three nodes; $X_1, X_2$ independent WGN($P_1$), WGN($P_2$) sources
- Two rounds: $1 \to 2$ followed by $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$

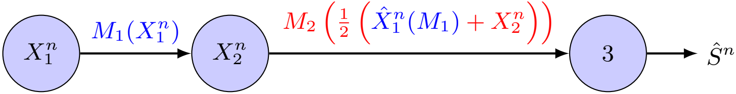


# 3-Node Cascade Network [Cuff, Su, EG 09]

- Three nodes; $X_1, X_2$ independent WGN($P_1$), WGN($P_2$) sources
- Two rounds: $1 \to 2$ followed by $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$

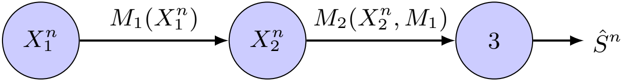



- Achievability schemes:
  - *Compute–Compress:*

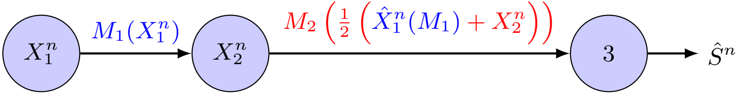


# 3-Node Cascade Network [Cuff, Su, EG 09]

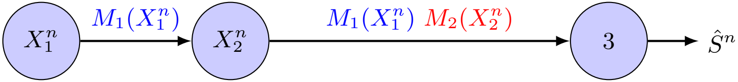
- Three nodes; $X_1, X_2$ independent WGN($P_1$), WGN($P_2$) sources
- Two rounds: $1 \to 2$ followed by $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$




- Achievability schemes:
  - *Compute–Compress:*

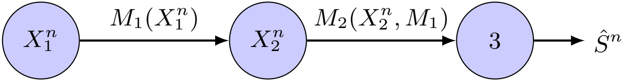



  - *Forward:*

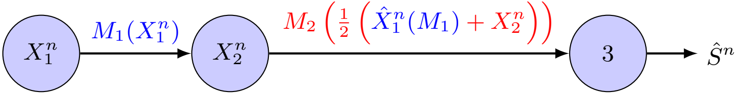


# 3-Node Cascade Network [Cuff, Su, EG 09]

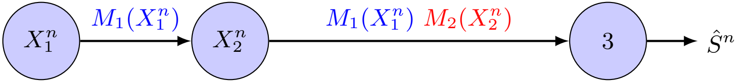
- Three nodes; $X_1, X_2$ independent WGN($P_1$), WGN($P_2$) sources
- Two rounds: $1 \to 2$ followed by $2 \to 3$
- Node 3 wishes to estimate $S^n = (X_1^n + X_2^n)/2$ with distortion $D$




- Achievability schemes:
  - *Compute–Compress:*



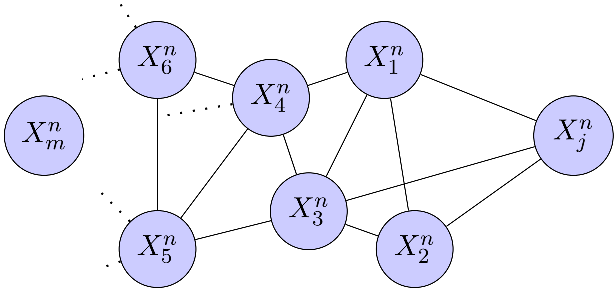

  - *Forward:*




- Neither strategy is optimal; problem remains open

# Cutset Lower Bound on $R^*(D)$

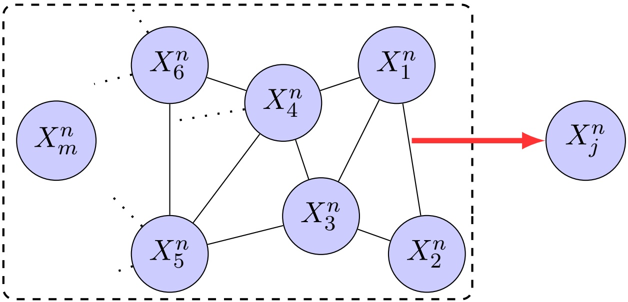
- Independent WGN(1) sources



# Cutset Lower Bound on $R^*(D)$

- Independent WGN(1) sources

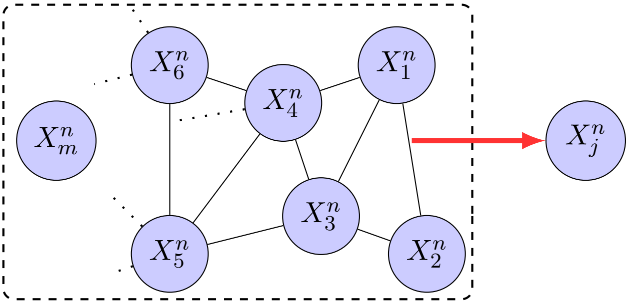
Super-node
$P = \frac{m-1}{m^2}$



# Cutset Lower Bound on $R^*(D)$

- Independent WGN(1) sources

Super-node
$P = \frac{m-1}{m^2}$

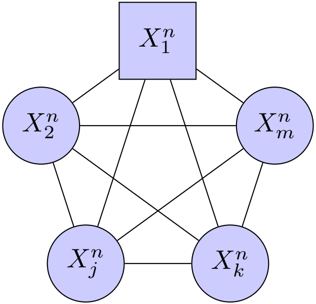



**Theorem (Su,EG 09)**

$$R^*(D) \geq \frac{m}{2} \log\left(\frac{m-1}{m^2 D}\right) \quad \textit{for } D < (m-1)/m^2$$

# Upper Bound on $R^*(D)$

- Use centralized protocol

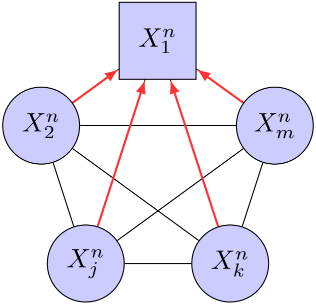


# Upper Bound on $R^*(D)$

- Use centralized protocol
- Round $t = 1, \ldots, (m-1)$:

$$r_j(t) = \frac{1}{2}\log\left(\frac{1}{d}\right)$$

$$d = \frac{mD}{2}$$



# Upper Bound on $R^*(D)$

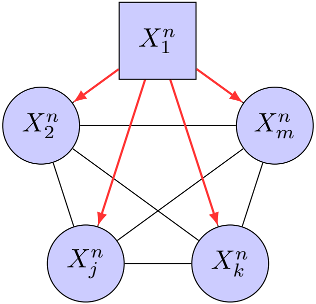
- Use centralized protocol
- Round $t = 1, \ldots, (m-1)$:

$$r_j(t) = \frac{1}{2}\log\left(\frac{1}{d}\right)$$

$$d = \frac{mD}{2}$$

- Round $t = m, \ldots, (2m-2)$:

$$r_1(t) = \frac{1}{2}\log\left(\frac{2}{mD}\right)$$



# Upper Bound on $R^*(D)$

- Use centralized protocol
- Round $t = 1, \ldots, (m-1)$:

$$r_j(t) = \frac{1}{2}\log\left(\frac{1}{d}\right)$$

$$d = \frac{mD}{2}$$

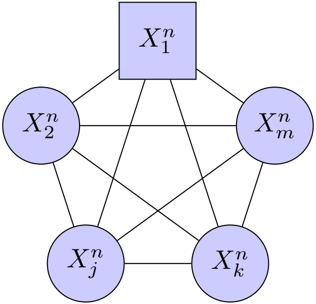
- Round $t = m, \ldots, (2m-2)$:

$$r_1(t) = \frac{1}{2}\log\left(\frac{2}{mD}\right)$$

- Upper bound:

$$R^*(D) \le (m-1)\log\left(\frac{2}{mD}\right)$$

Within 2x of cutset bound



# Distributed Weighted-Sum Protocols

- $T$ rounds of node-pair, two-way communication/computing
- Estimate of node $j$ at $t = 0$, $S_j^n(0) = X_j^n$

# Distributed Weighted-Sum Protocols

- $T$ rounds of node-pair, two-way communication/computing
- Estimate of node $j$ at $t = 0$, $S_j^n(0) = X_j^n$
- Assume $(j, k)$ selected in round $t > 0$:
  - Node $j$: Sends description $\hat{S}_j^n(t)$ to node $j$
    (Distortion $d\,\mathrm{E}(S_j^2(t))$ and *rate* $(1/2)\log(1/d)$)
  - Similarly, node $k$ sends description $\hat{S}_k^n(t)$ to node $j$

# Distributed Weighted-Sum Protocols

- $T$ rounds of node-pair, two-way communication/computing
- Estimate of node $j$ at $t = 0$, $S_j^n(0) = X_j^n$
- Assume $(j, k)$ selected in round $t > 0$:
  - Node $j$: Sends description $\hat{S}_j^n(t)$ to node $j$ (Distortion $d\,\mathrm{E}(S_j^2(t))$ and *rate* $(1/2)\log(1/d)$)
  - Similarly, node $k$ sends description $\hat{S}_k^n(t)$ to node $j$
  - Nodes update their estimates

$$S_v^n(t+1) = \frac{1}{2} S_v^n(t) + \frac{1}{2(1-d)} \hat{S}_{j+k-v}^n(t) \text{ for } v = j, k$$

# Distributed Weighted-Sum Protocols

- $T$ rounds of node-pair, two-way communication/computing
- Estimate of node $j$ at $t = 0$, $S_j^n(0) = X_j^n$
- Assume $(j, k)$ selected in round $t > 0$:
  - Node $j$: Sends description $\hat{S}_j^n(t)$ to node $j$
    (Distortion $d\,\mathrm{E}(S_j^2(t))$ and *rate* $(1/2)\log(1/d)$)
  - Similarly, node $k$ sends description $\hat{S}_k^n(t)$ to node $j$
  - Nodes update their estimates

$$S_v^n(t+1) = \frac{1}{2} S_v^n(t) + \frac{1}{2(1-d)} \hat{S}_{j+k-v}^n(t) \text{ for } v = j, k$$

- Define $R_{\mathrm{WS}}^*(D) \geq R^*(D)$

# Distributed Weighted-Sum Protocols

- $T$ rounds of node-pair, two-way communication/computing
- Estimate of node $j$ at $t = 0$, $S_j^n(0) = X_j^n$
- Assume $(j, k)$ selected in round $t > 0$:
  - Node $j$: Sends description $\hat{S}_j^n(t)$ to node $j$ (Distortion $d\,\mathrm{E}(S_j^2(t))$ and *rate* $(1/2)\log(1/d)$)
  - Similarly, node $k$ sends description $\hat{S}_k^n(t)$ to node $j$
  - Nodes update their estimates

$$S_v^n(t+1) = \frac{1}{2} S_v^n(t) + \frac{1}{2(1-d)} \hat{S}_{j+k-v}^n(t) \text{ for } v = j, k$$

- Define $R_{\mathrm{WS}}^*(D) \geq R^*(D)$

**Theorem (Lower Bound on $R_{\mathrm{WS}}^*(D)$)**

$$R_{\mathrm{WS}}^*(D) \geq \left( \frac{m}{2} \log \frac{1}{\sqrt{D} + 1/m} \right) \left( \log \frac{1}{4mD} \right)$$

# Gossip-Based Weighted-Sum Protocol

- Node-pair selected independently at random in each round

# Gossip-Based Weighted-Sum Protocol

- Node-pair selected independently at random in each round

- Expected weighted-sum network rate distortion function

$$\mathrm{E}(R_{\mathsf{WS}}(D)) = \inf \{\mathrm{E}(R) : (R, \Delta) \text{ is achievable}, \mathrm{E}(\Delta) \le D\}$$

# Gossip-Based Weighted-Sum Protocol

- Node-pair selected independently at random in each round

- Expected weighted-sum network rate distortion function

$$\mathrm{E}(R_{\mathsf{WS}}(D)) = \inf \{\mathrm{E}(R) : (R, \Delta) \text{ is achievable}, \mathrm{E}(\Delta) \le D\}$$

- $R^*_{\mathrm{WS}}(D) \le \mathrm{E}(R_{\mathrm{WS}}(D))$

# Gossip-Based Weighted-Sum Protocol

- Node-pair selected independently at random in each round
- Expected weighted-sum network rate distortion function

$$\mathrm{E}(R_{\mathsf{WS}}(D)) = \inf\{\mathrm{E}(R) : (R, \Delta) \text{ is achievable}, \mathrm{E}(\Delta) \le D\}$$

- $R^*_{\mathrm{WS}}(D) \le \mathrm{E}(R_{\mathrm{WS}}(D))$

**Theorem (Upper bound on $\mathrm{E}(R_{\mathrm{WS}}(D))$)**

$$\mathrm{E}\left(R_{WS}(D)\right) \le (m-1)\left(\ln\frac{2}{D}\right)\left(\log\frac{(m-1)\ln(2/D)}{m^2 D}\right)$$

# Summary of Bounds

- Complete graph

# Summary of Bounds

- Complete graph

| Bounds | $D = \Theta(1/m)$ | $D = \Theta(1/m^2)$ |
| --- | --- | --- |

# Summary of Bounds

- Complete graph

| | Bounds | $D = \Theta(1/m)$ | $D = \Theta(1/m^2)$ |
|---|---|---|---|
| Centralized | $R^*(D)$ | $\Theta(m)$ | $\Theta(m \log m)$ |

# Summary of Bounds

- Complete graph

| | Bounds | $D = \Theta(1/m)$ | $D = \Theta(1/m^2)$ |
|---|---|---|---|
| Centralized | $R^*(D)$ | $\Theta(m)$ | $\Theta(m \log m)$ |
| Distributed | $R^*_{\mathsf{WS}}(D)$ | $\Omega(m \log m)$ | $\Omega(m(\log m)^2)$ |
| | $\mathrm{E}\left(R_{\mathsf{WS}}(D)\right)$ | $O(m(\log m)(\log\log m))$ | $O(m(\log m)^2)$ |

# Summary of Bounds

- Complete graph

| | Bounds | $D = \Theta(1/m)$ | $D = \Theta(1/m^2)$ |
|---|---|---|---|
| Centralized | $R^*(D)$ | $\Theta(m)$ | $\Theta(m \log m)$ |
| Distributed | $R^*_{\mathsf{WS}}(D)$ | $\Omega(m \log m)$ | $\Omega(m(\log m)^2)$ |
| | $\mathrm{E}\left(R_{\mathsf{WS}}(D)\right)$ | $O(m(\log m)(\log\log m))$ | $O(m(\log m)^2)$ |

Penalty of using distributed protocol is factor of $\log m$ in sum rate

# Conclusion

- Distributed lossy averaging as example of lossy distributed computing
  - Upper and lower bounds on network rate distortion function
  - Formulation allowed us to quantify the penalty of distributedness
  - Network rate distortion function not known in general

# Conclusion

- Distributed lossy averaging as example of lossy distributed computing
  - Upper and lower bounds on network rate distortion function
  - Formulation allowed us to quantify the penalty of distributedness
  - Network rate distortion function not known in general

- To learn more:
  - Su, EG, "Distributed Lossy Averaging," http://arxiv.org/abs/0901.4134
  - Cuff, Su, EG, "Cascade Multiterminal Source Coding," http://arxiv.org/abs/0905.1883
  - Kim, EG, "Lecture notes on network information theory," http://arxiv.org/abs/1001.3404

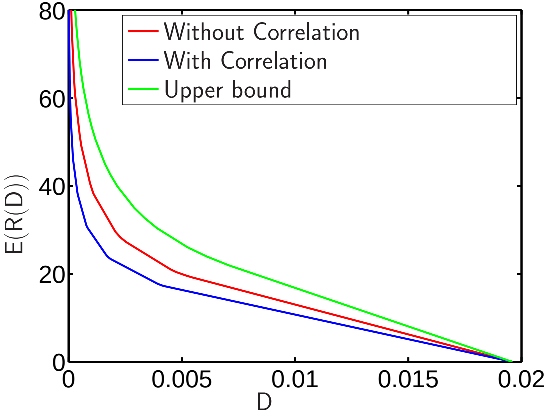
# Thank You

# Effect of Using Correlation

- We ignored the build up in correlation
- Can achieve better rate using Wyner-Ziv coding
- Very difficult to analyze

# Effect of Using Correlation

- We ignored the build up in correlation
- Can achieve better rate using Wyner-Ziv coding
- Very difficult to analyze
- Using simulations ($m = 50$):



# Effect of Using Correlation

- We ignored the build up in correlation
- Can achieve better rate using Wyner-Ziv coding
- Very difficult to analyze
- Using simulations ($m = 50$):