Group Gossip: Distributed Consensus Through Multilateral Wireless Exchanges

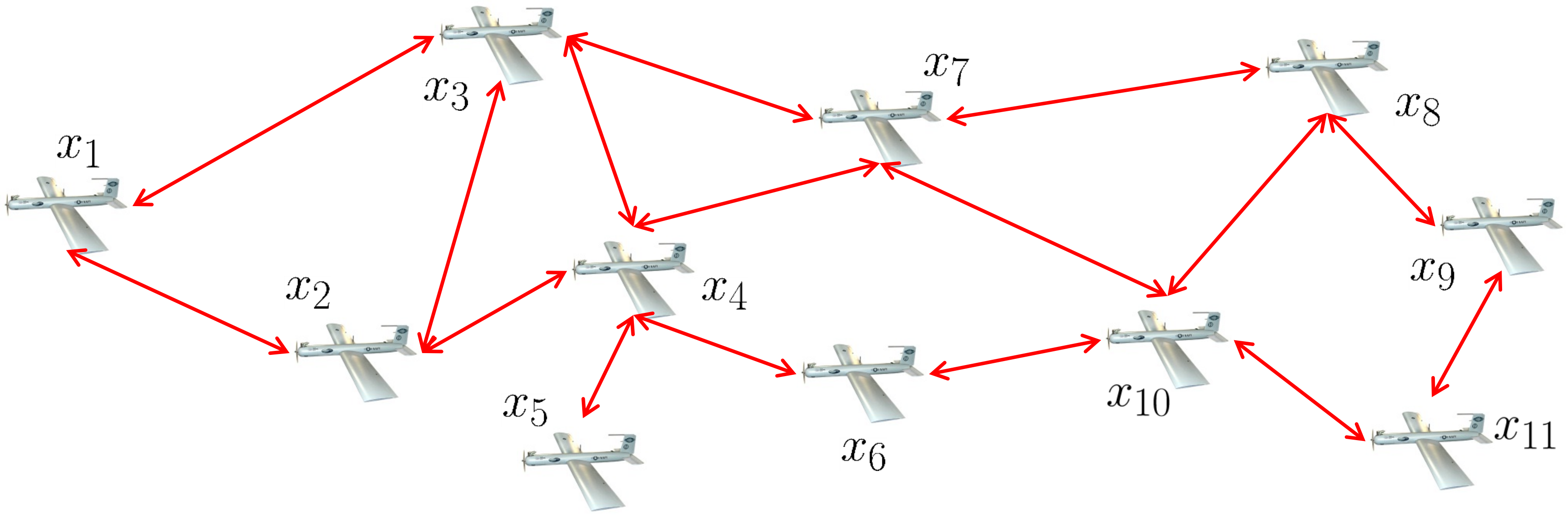
Network Mapping and Measurements Conference 2012
March 20, 2012

Waheed U. Bajwa
Electrical and Computer Engineering, Rutgers University

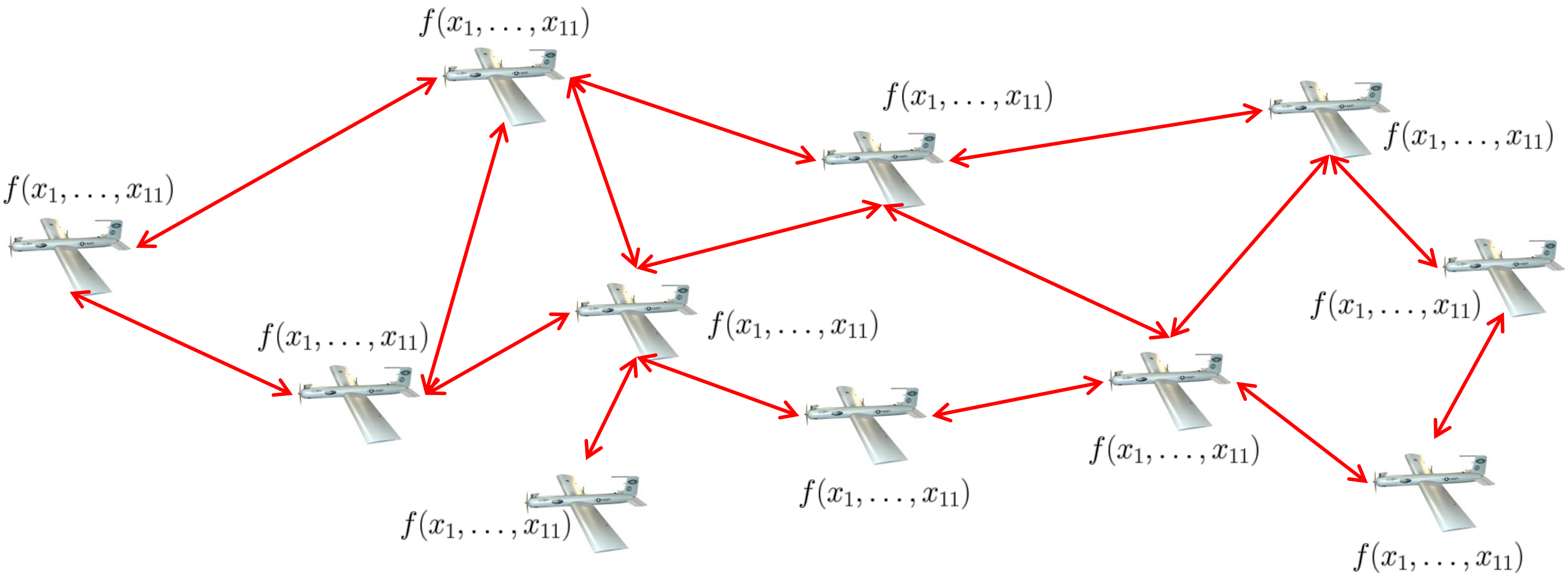
Joint work with
Matthew Nokleby and Behnaam Aazhang (Rice University)
Robert Calderbank (Duke University)


THE MAGAZINE THE STARS TRUST
OK!
KANYE & KIM
YES, THEY'RE BACK TOGETHER
JUSTIN & JEN
SUDDEN SPLIT
IT'S OFFICIAL! WEDDING OF THE YEAR
FINALLY...
MRS. BRAD PITT!
Angie agrees to wed as the couple gets ready for more kids
THE HUNGER GAMES
SURPRISE WEDDING!
JESSICA VS NICK
IT'S A BABY SHOWDOWN!
ALL THE EXCLUSIVE DETAILS!

Distributed Consensus



Distributed Consensus



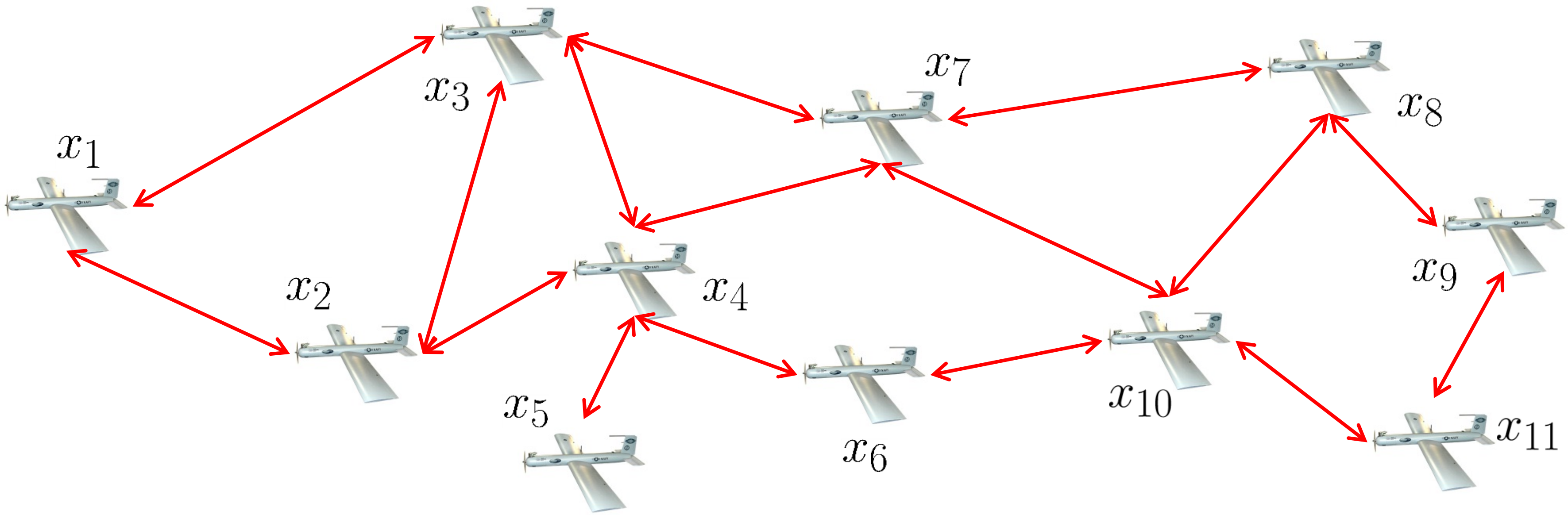
Distributed Consensus

- Compute a function $f(x_1, x_2, \ldots, x_N)$ of initial values in a distributed fashion at every node
- Goals: Robustness, guaranteed convergence, and fast rate of convergence

This talk: Distributed ***linear*** consensus, i.e., $f(x_1, x_2, \ldots, x_N) = \sum_{i=1}^{N} a_i x_i$

Related references: Tsitsiklis’84, Rabani et al.’98, Xiao-Boyd’03, Jadbabaie et al.’03, etc.

Gossip Algorithms for Distributed Consensus



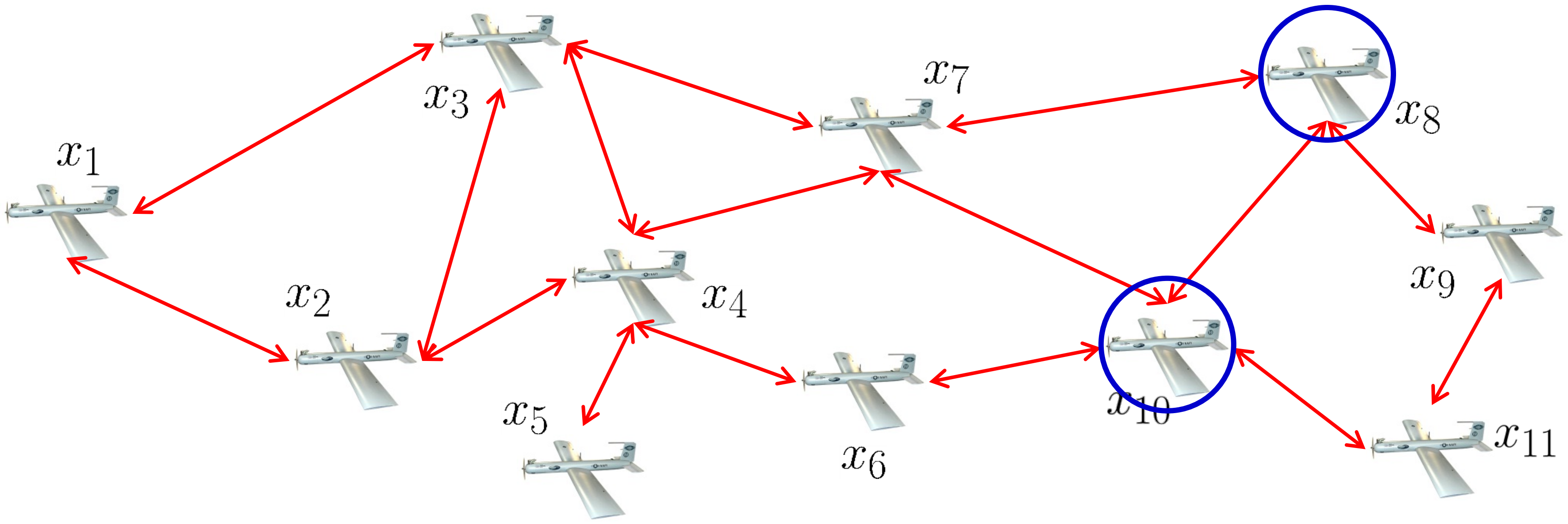
Basic premise: Only pairwise exchanges take place in the network

Randomized Gossip

- Pairwise exchange of existing estimates between randomly-chosen neighbors

Related references: Karp et al.'00, Kempe-Kleinberg'02, Boyd et al.'06, etc.

Gossip Algorithms for Distributed Consensus



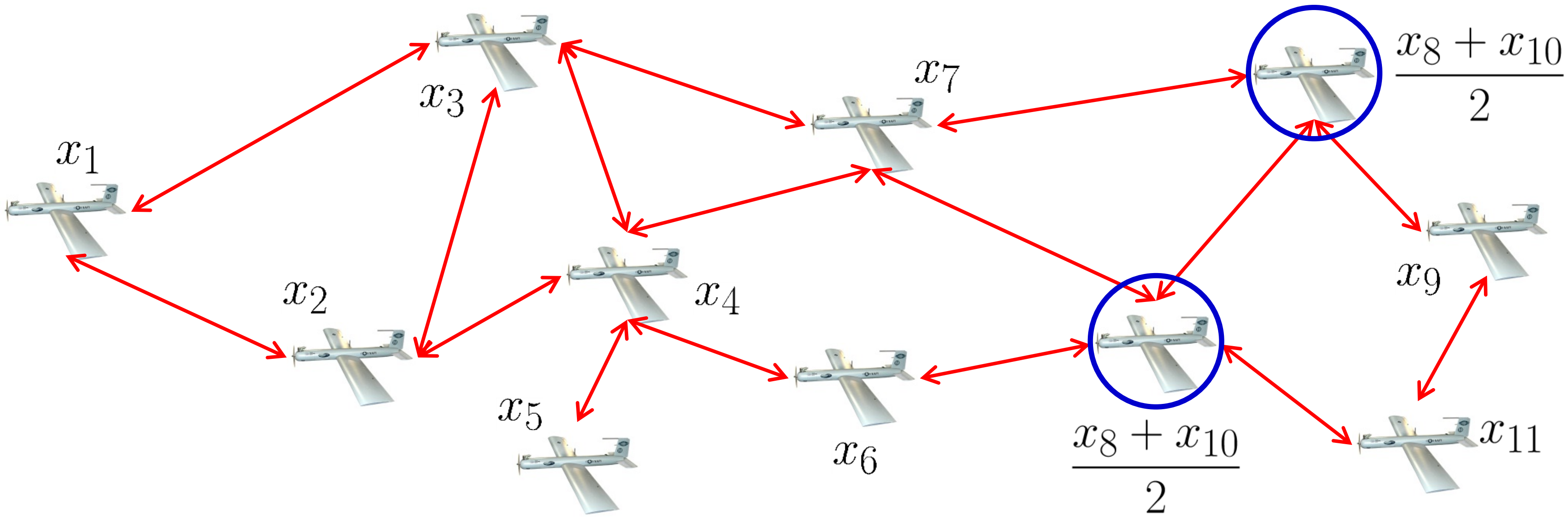
Basic premise: Only pairwise exchanges take place in the network

Randomized Gossip

- Pairwise exchange of existing estimates between randomly-chosen neighbors

Related references: Karp et al.'00, Kempe-Kleinberg'02, Boyd et al.'06, etc.

Gossip Algorithms for Distributed Consensus



Basic premise: Only pairwise exchanges take place in the network

Randomized Gossip

- Pairwise exchange of existing estimates between randomly-chosen neighbors
- Update the estimate at each node by averaging the two estimates

Related references: Karp et al.'00, Kempe-Kleinberg'02, Boyd et al.'06, etc.

Gossip Algorithms for Distributed Consensus

Random pairwise exchanges lead to random node dynamics

$$\mathbf{x}(t) = \mathbf{W}(t)\mathbf{x}(t-1)$$

- $\mathbf{W}(t)$ is a matrix describing the estimate updates
- $W_{ij}(t) = 0$ if nodes i and j are not neighbors

Existing literature: Various gossip algorithms have favorable convergence times

- Randomized gossip of Boyd et al.'06
- Geographic gossip of Dimakis et al.'06
- Geographic gossip with path averaging of Benezit et al.'07
- Multiscale gossip of Tsianos-Rabbat'10
- and some others …

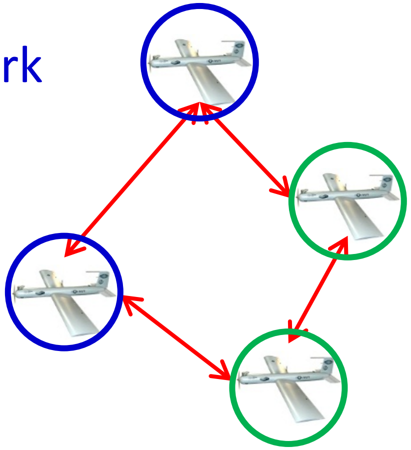
How do messages get exchanged in gossip algorithms?

They really do not ...

- Graphical topology abstracts away the physical and medium access layers

But distributed consensus usually takes place in a wireless network

- Challenges: Interference, retransmissions, and hidden nodes
- Opportunities: Broadcast and superposition inherent in wireless



State-of-the-Art Gossip Algorithms

- Ignore overhead of interference and collisions
  - Performances are bound to be worse than the advertised ones
- Do not exploit the broadcast and superposition properties of wireless
  - Exceptions: Aysal et al.'09 (broadcast) and Nazer et al.'11 (superposition)

Distributed Consensus in Wireless Networks

Algorithmic Objectives

- Explicitly account for the interference inherent in wireless networks
- Exploit the superposition and broadcast nature of the wireless medium
- Characterize performance: Does the performance worsen or get better?

Our approach to distributed consensus in wireless networks: Combination of …

- Superposition and broadcast properties
- Sparse signal recovery techniques
- Consensus algorithms (analytical tools)

Distributed Consensus Through Multilateral Wireless Exchanges

Wireless Framework

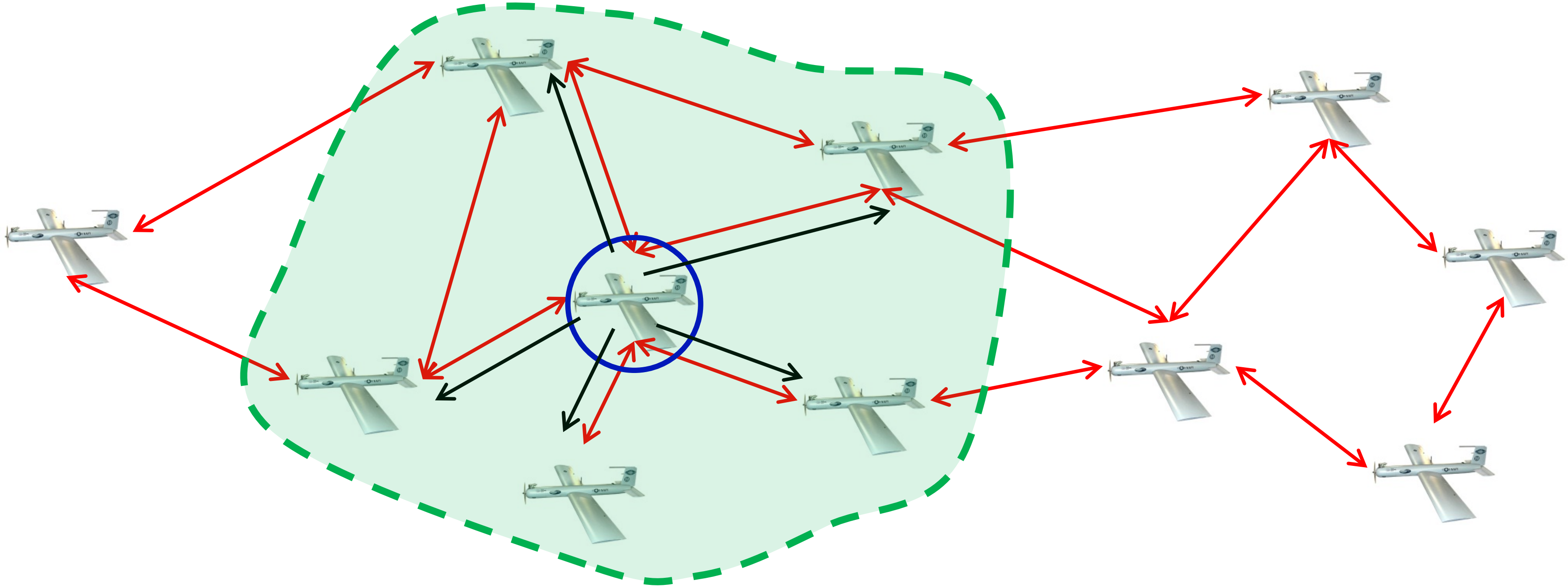
- Nodes use on-off keying to "virtually" transmit/receive simultaneously (Zhang-Guo'10)
- Nodes use "virtual full-duplex" to transmit to/receive from all neighbors simultaneously
- Sparse recovery techniques guarantee robust multiuser detection at every node

Enables "Group Gossip"

- Nodes exchange their current estimates with multiple neighbors
- Analysis (and simulations) offer 3x improvement in convergence rate
- "Order improvement" in convergence rate is possible in certain networks

Interference Management via Virtual Full-Duplex

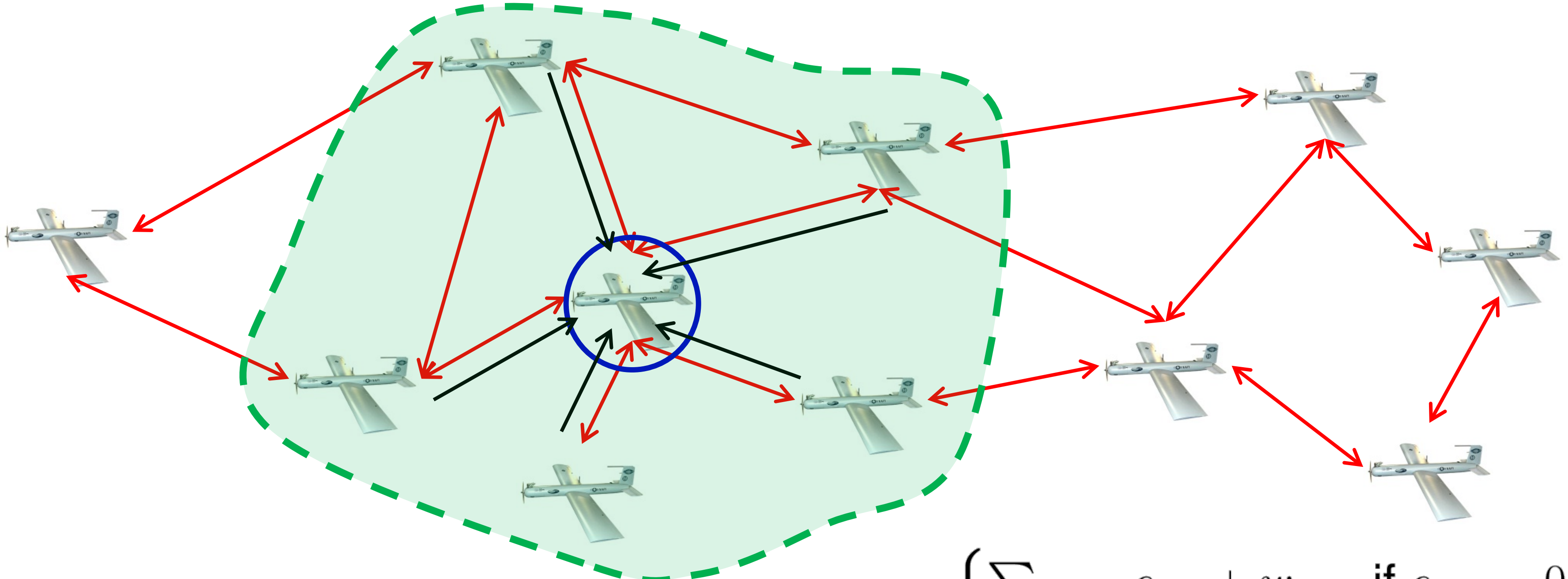
- Wireless network with slotted time, global clock, and fixed bandwidth
- Nodes simultaneously transmit/receive codewords comprising {0, 1, -1} symbols
- Interference at a node is caused by transmissions arriving from every neighbor



Zhang and Guo, "Wireless peer-to-peer mutual broadcast via sparse recovery," 2010

Interference Management via Virtual Full-Duplex

- Wireless network with slotted time, global clock, and fixed bandwidth
- Nodes simultaneously transmit/receive codewords comprising {0, 1, -1} symbols
- Interference at a node is caused by transmissions arriving from every neighbor
- RF Hardware: Half-duplex, i.e., a node can only receive when "transmitting" 0 symbol



$$y_{j,n} = \begin{cases} \sum_{i \in \mathcal{N}_i} c_{i,n} + w_j, & \text{if } c_{j,n} = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Interference Management via Virtual Full-Duplex

- Each node in the network is assigned 2^L codewords of length M

$$\mathbf{y}_j = \begin{bmatrix} 0 & 1 & 1 & -1 & 0 & 1 & 0 & 0 & -1 & 1 & 0 & -1 \\ 1 & -1 & 0 & 0 & -1 & 1 & 1 & 0 & 1 & 0 & 0 & -1 \\ -1 & -1 & 0 & 1 & 0 & 0 & 1 & -1 & 0 & -1 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & 1 & 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 0 & -1 & 1 & -1 & 0 & -1 & 1 & -1 \\ 1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{w}_j$$

Zhang and Guo, "Wireless peer-to-peer mutual broadcast via sparse recovery," 2010

Interference Management via Virtual Full-Duplex

- Each node in the network is assigned 2^L codewords of length M

$$\mathbf{y}_j = \begin{bmatrix} 0 & 1 & 1 & -1 & 0 & 1 & 0 & 0 & -1 & 1 & 0 & -1 \\ \multicolumn{12}{c}{\text{[redacted]}} \\ 0 & 1 & -1 & 0 & 0 & 1 & 1 & -1 & 1 & -1 & 0 & 0 \\ \multicolumn{12}{c}{\text{[redacted]}} \\ 1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{w}_j$$

Interference management reduces to sparse signal recovery

- Can be solved using convex optimization (Applebaum et al.'10 & '12)
- In the case of random codewords, it requires (Nokleby et al.'11)

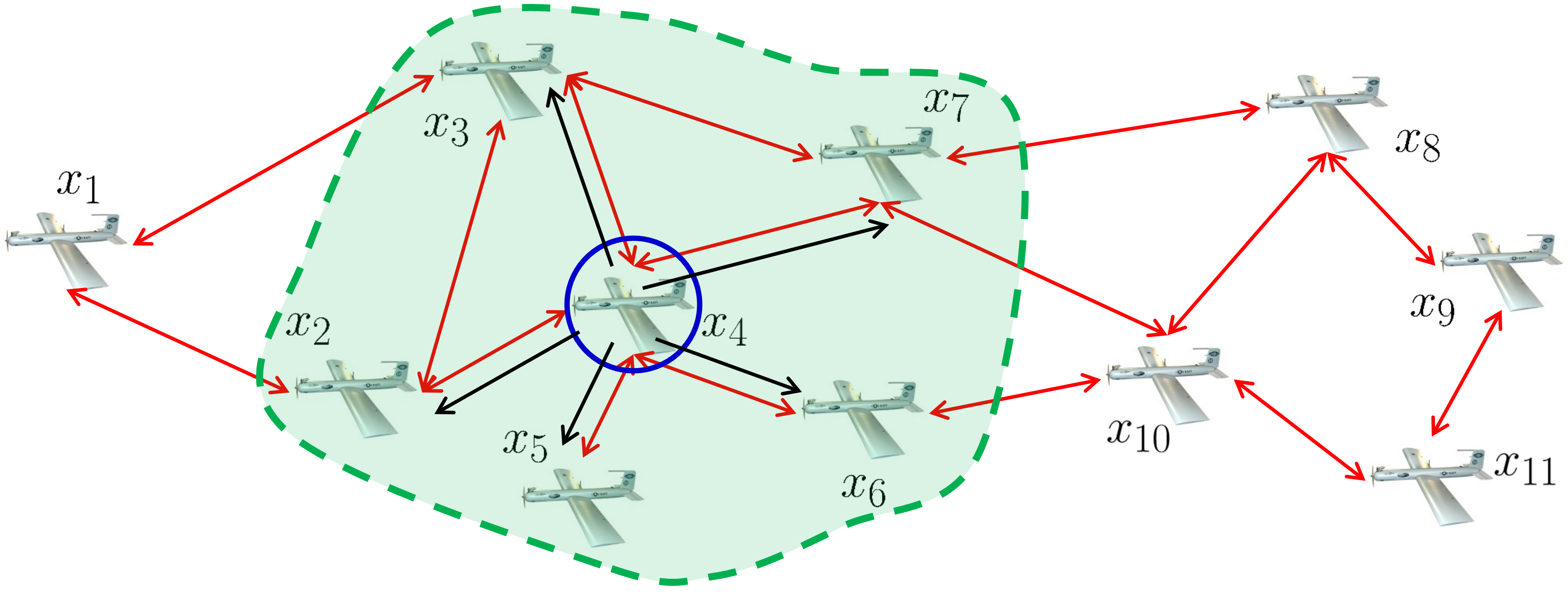
$$M = O\left(\max\left\{L|\mathcal{N}_j|, \log(N)\right\}\right)$$

- Deterministic codewords: Preliminary work reported in Applebaum et al.'11

Group Gossip for Distributed Consensus

Multilateral exchanges take place between a node and all its neighbors

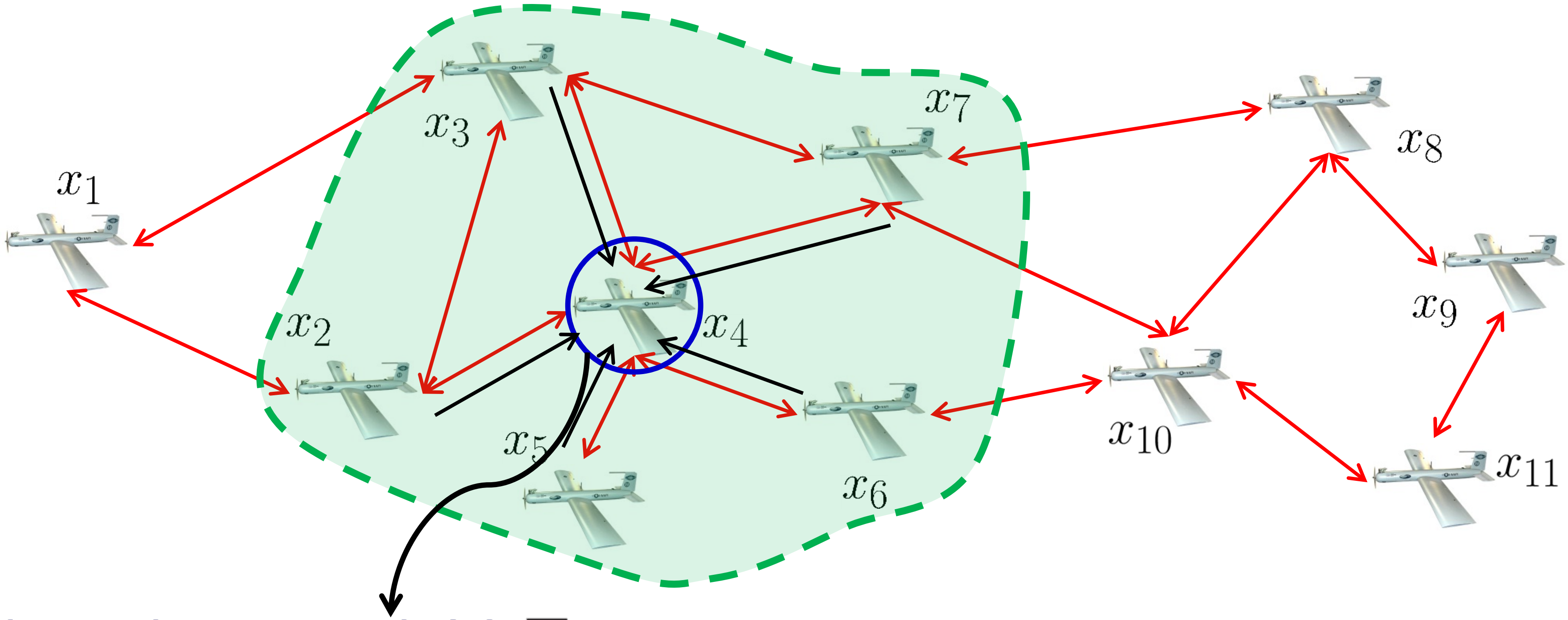
- Each node broadcasts its estimate via on-off codewords



Group Gossip for Distributed Consensus

Multilateral exchanges take place between a node and all its neighbors

- Each node broadcasts its estimate via on-off codewords
- Simultaneously, the node decodes its neighbors' estimates using, e.g., the lasso
- The node finally takes a convex combination of its neighbors' estimates (Xiao-Boyd'03)



New estimate at node '4': $\sum_{j \in \mathcal{N}_4} w_{4j} x_j$

Group Gossip for Distributed Consensus

Deterministic multilateral exchanges lead to deterministic node dynamics

$$\mathbf{x}(t) = \mathbf{W}\mathbf{x}(t-1)$$

- $\mathbf{W}$ is a matrix describing the convex estimate updates
- $W_{ij} = 0$ if nodes i and j are not neighbors

We need to specify the convergence speed of group gossip, which we define

$$T_{\text{ave}}(\mathbf{W}, \epsilon) = \sup_{\mathbf{x}(0) \in \mathbb{R}^N} \inf \left\{ t : \Pr\left(\frac{\|\mathbf{x}(t) - x_{\text{ave}}\mathbf{1}\|}{\|\mathbf{x}(0)\|} \geq \epsilon \right) \leq \epsilon \right\}$$

In words, the time it takes to achieve some prescribed error ϵ in the network for the worst-case initial network data $\mathbf{x}(0)$

Convergence Speed of Group Gossip

Theorem [Nokleby, ***B.***, Calderbank, and Aazhang'11]

The ϵ-averaging time of group gossip can be bounded using the second-largest eigenvalue $\lambda_2(\mathbf{W})$ of the positive semi-definite mixing matrix $\mathbf{W}$ as follows:

$$\frac{\log(\epsilon^{-1}) - \log(\sqrt{2})}{\log(\lambda_2(\mathbf{W})^{-1})} \leq T_{\text{ave}}(\mathbf{W}, \epsilon) \leq \frac{\log(\epsilon^{-1})}{\log(\lambda_2(\mathbf{W})^{-1})}.$$

Convergence Speed of Group Gossip

Theorem [Nokleby, ***B.***, Calderbank, and Aazhang'11]

The ϵ-averaging time of group gossip can be bounded using the second-largest eigenvalue $\lambda_2(\mathbf{W})$ of the positive semi-definite mixing matrix $\mathbf{W}$ as follows:

$$\frac{\log(\epsilon^{-1}) - \log(\sqrt{2})}{\log(\lambda_2(\mathbf{W})^{-1})} \leq T_{\text{ave}}(\mathbf{W}, \epsilon) \leq \frac{\log(\epsilon^{-1})}{\log(\lambda_2(\mathbf{W})^{-1})}.$$

Comparison: Randomized Gossip

- The mixing matrix $\mathbf{W}(t)$ is drawn iid from a symmetric, doubly-stochastic matrix $\mathbf{P}$
- Averaging time is described using the second-largest eigenvalue of $\mathbf{P}$

$$\frac{\log(\epsilon^{-1})}{2\log(\lambda_2(\mathbf{P})^{-1})} \leq T_{\text{ave}}(\mathbf{P}, \epsilon) \leq \frac{3\log(\epsilon^{-1})}{\log(\lambda_2(\mathbf{P})^{-1})}$$

Algorithmic Advantages

- Set of $\mathbf{W}$ matrices (group gossip) is larger than the set of $\mathbf{P}$ matrices (randomized gossip)
- Tighter bounds on the averaging time, with 3x improvement on the upperbound

Finding the Optimal Mixing Matrix

Finding the group gossip algorithm with smallest averaging time is equal to …

- Finding the mixing matrix $\mathbf{W}$ with smallest second-largest eigenvalue

In our setup, this can be cast as a convex program …

$$
\begin{aligned}
\min_{\mathbf{W}\succeq 0 \in \mathbb{R}^{N\times N}} \quad & \sigma_2(\mathbf{W}) \\
\text{subject to} \quad & \mathbf{W}\mathbf{1} = \mathbf{1} \\
& \mathbf{1}^T\mathbf{W} = \mathbf{1}^T \\
& w_{ij} \geq 0 \\
& w_{ij} = 0 \text{ for } (i,j) \text{ not neighbors}
\end{aligned}
$$

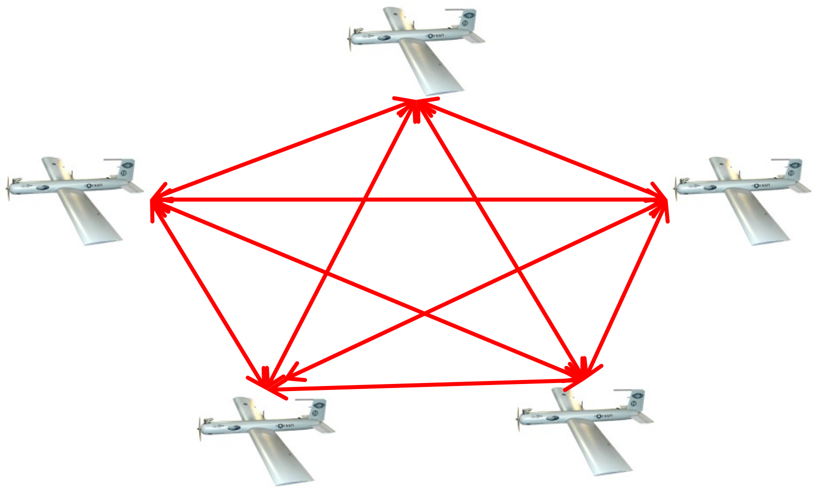
… which can be solved using a subgradient projection method, among others

Other references: Xiao-Boyd’03

Group Gossip Over Specific Network Topologies

Fully-Connected Network

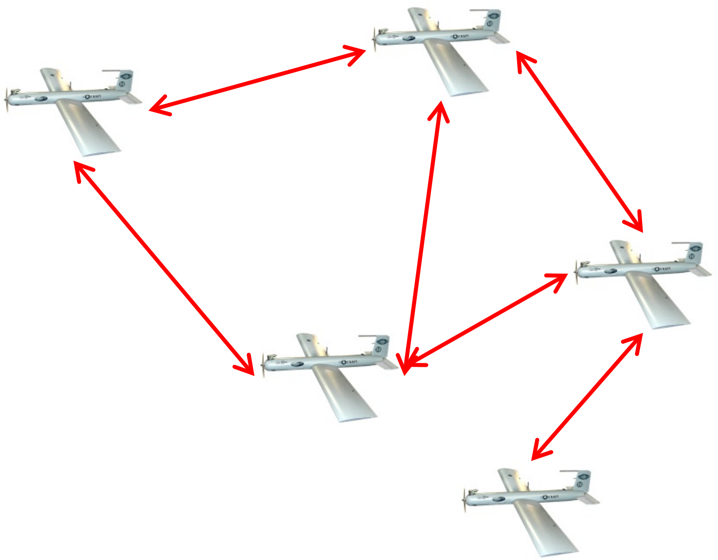
- Group Gossip: $T_{\text{ave}}(\mathbf{W}, \epsilon) = 1$
- Randomized Gossip: $T_{\text{ave}}(\mathbf{P}, \epsilon) = \log(\epsilon^{-1})$



Random-Geographic Network (Unit Square)

- Group and randomized gossip have the same rate

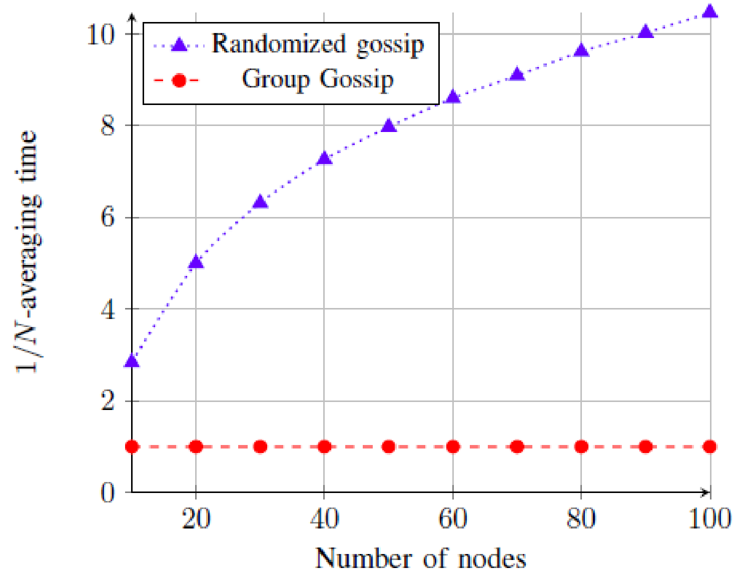
$$T_{\text{ave}}(\mathbf{W}, \epsilon) = O\left(\frac{N \log(\epsilon^{-1})}{\log(N)}\right)$$



Numerical Results (SNR = 10 dB, 16 bit quantization, $\epsilon = 1/N$)

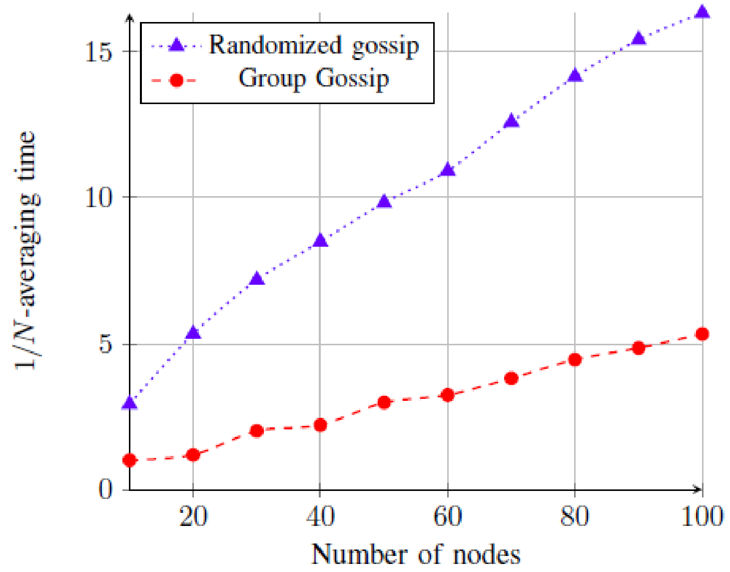
Fully-Connected Network

- Group gossip converges in one round



Random-Geographic Network

- Group gossip gives 3x improvement



(Randomized gossip simulated without PHY/MAC)

Can we do even better?

Distributed consensus usually takes place in a wireless network

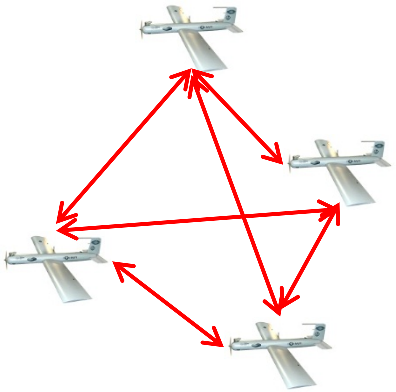
- Challenges: Interference, retransmissions, and hidden nodes
- Opportunities: Broadcast and superposition inherent in wireless

But we are still not exploiting one key property of wireless networks …

- Manipulating the topology of a network through power control

Sneak preview: Hierarchical group gossip

*(Nokleby, **B.**, Calderbank, and Aazhang'12)*



Conclusions

Distributed Consensus (a la Gossip Algorithms)

- Traditional literature typically ignores the PHY/MAC issues of wireless networks

Group Gossip

- An attempt to address the interference inherent in wireless networks
- Exploits broadcast and superposition properties of wireless to manage interference
- Performs at least 3x better than traditional gossip in spite of the practical considerations

What Next?

- Flexible topologies through power control for improved distributed consensus

Gossip is good, but group gossip is even better …