Distributed estimation and consensus

Luca Schenato
University of Padova
WIDE'09 7 July 2009, Siena

Joint work w/

Alessandro Chiuso
University of Padova

Sandro Zampieri
University of Padova

Ruggero Carli
UC Santa Barbara

Damiano Varagnolo
Univ. of Padova

Simone Del favero
Univ. of Padova

Saverio Bolognani
Univ. of Padova

Outline

- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
  - Distributed Kalman filtering
- Open problems
  - Identification
  - Estimation
  - Control

Outline

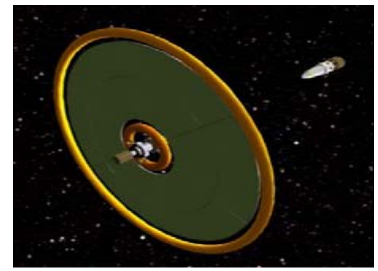
- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
  - Distributed Kalman filtering
- Open problems
  - Identification
  - Estimation
  - Control

Networked Control Systems

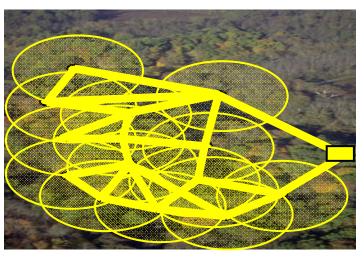
Drive-by-wire systems

Swarm robotics

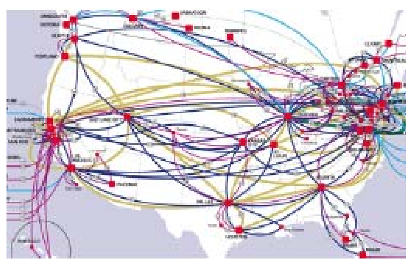
Smart structures:
space telescope & satellites mesh



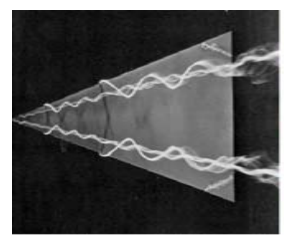
Wireless Sensor Networks

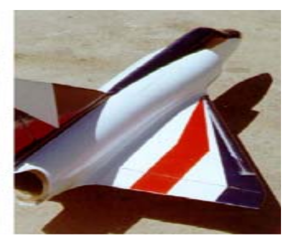


Traffic Control:
Internet and transportation



Smart materials & MEMS:
sheets of sensors and actuators

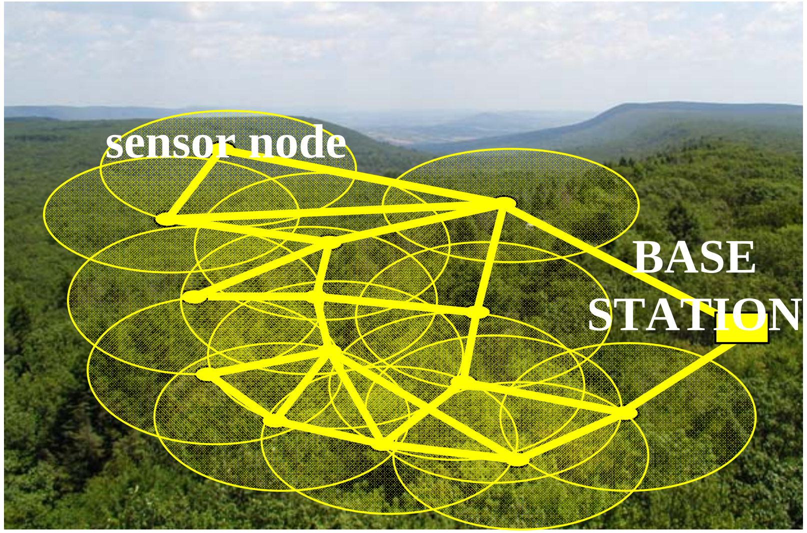




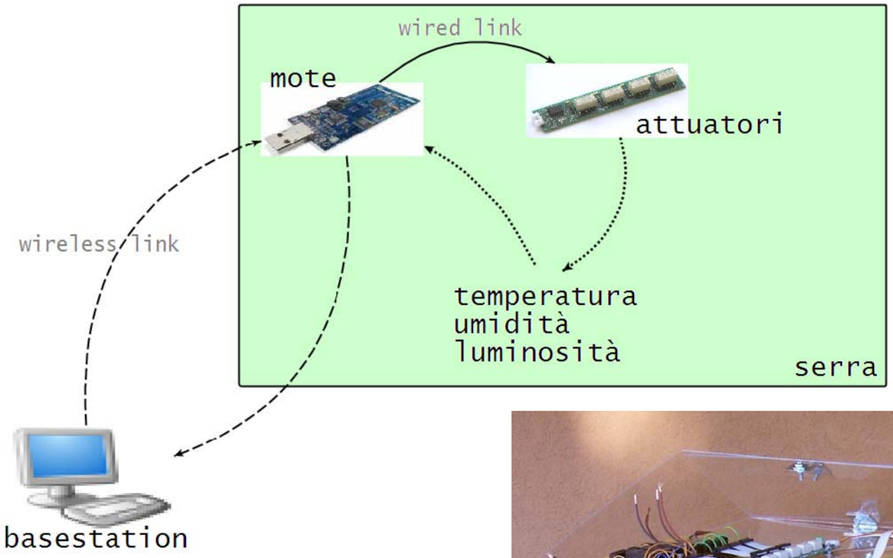
NCSs: physically distributed dynamical systems interconnected by a communication network

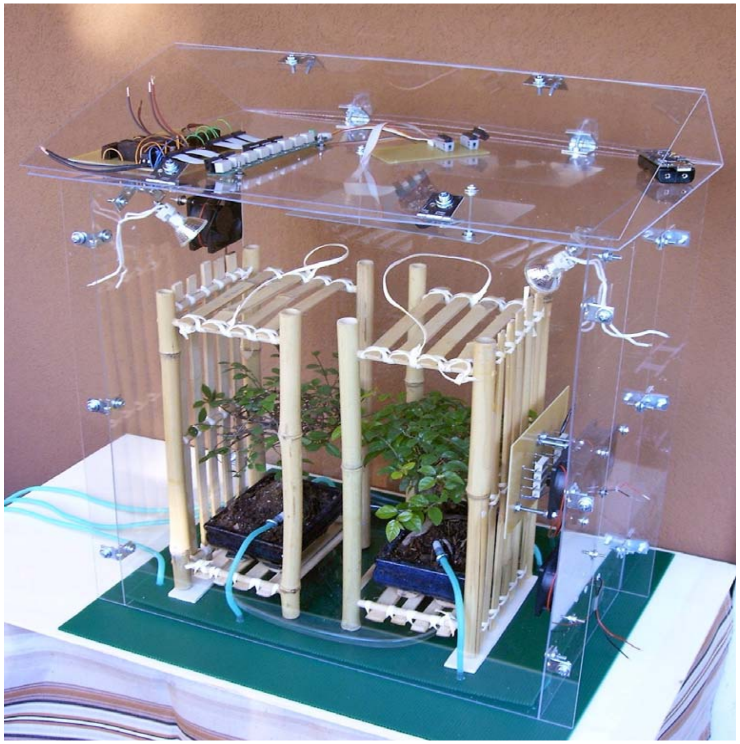
Wireless Sensor Actuator Networks (WSANs)

- Small devices
  - μController, Memory
  - Wireless radio
  - Sensors & Actuators
  - Batteries
- Inexpensive
- Multi-hop communication
- Programmable (micro-PC)



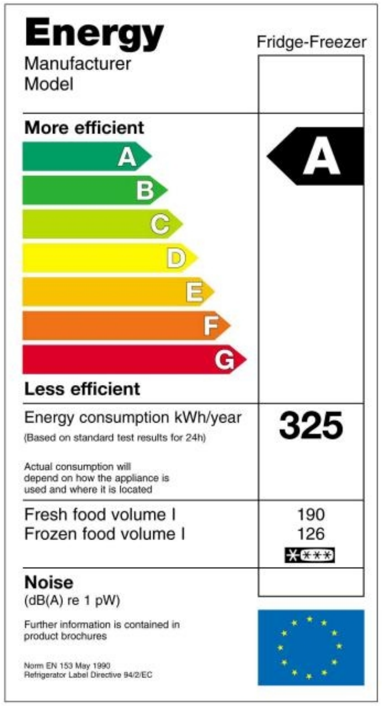
Applications: Smart Greenhouse





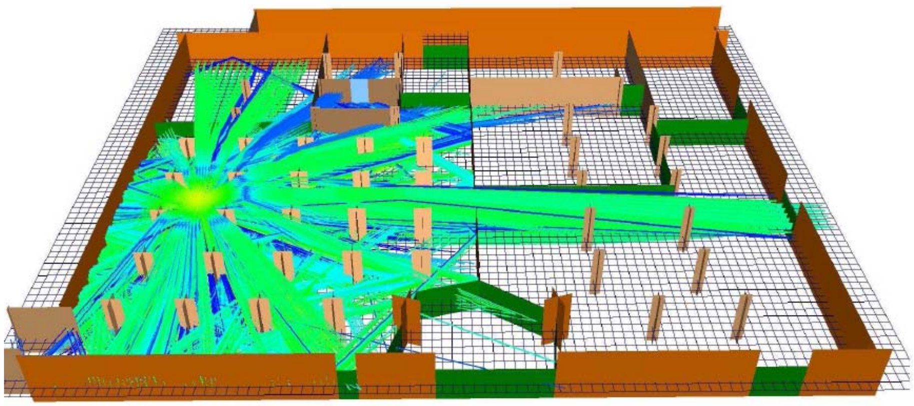
- Distributed estimation
- Distributed control
- Control under packet loss & random delay
- Sensor fusion
- Distributed time synchronization

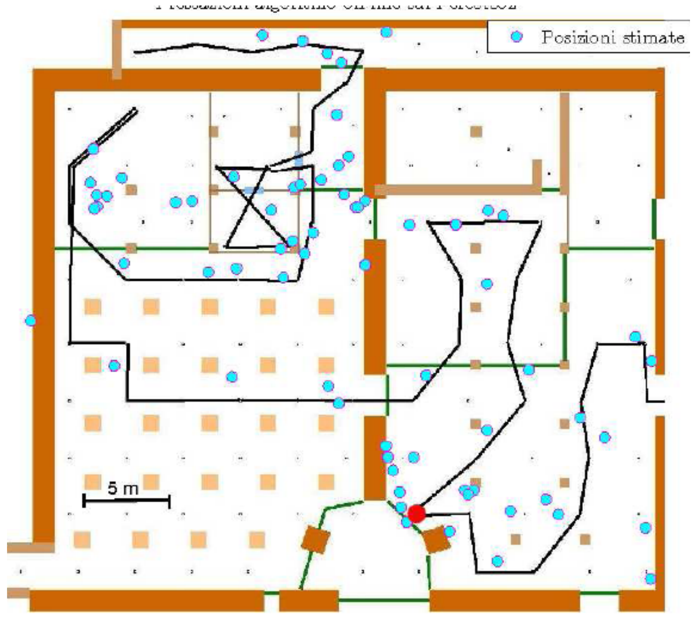
Applications: ThermoEfficiency Labeling



- Building thermodynamics model identification
- Sensor selection for identification
- Optimal sensor placement

Applications:
Distributed Localization&Tracking

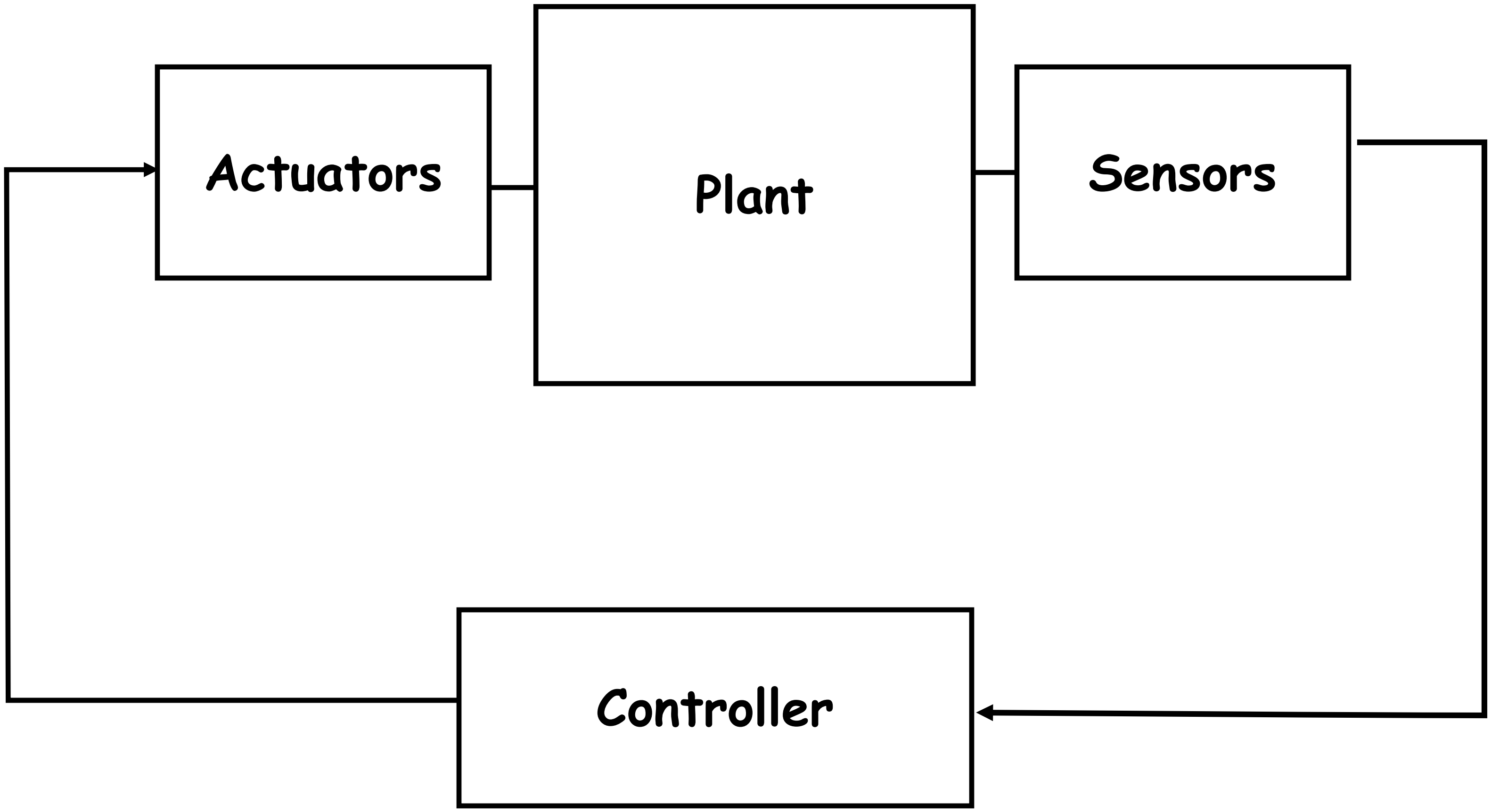




- Indoor radio signal modeling
- Real-time localization
- Distributed tracking
- Coordination

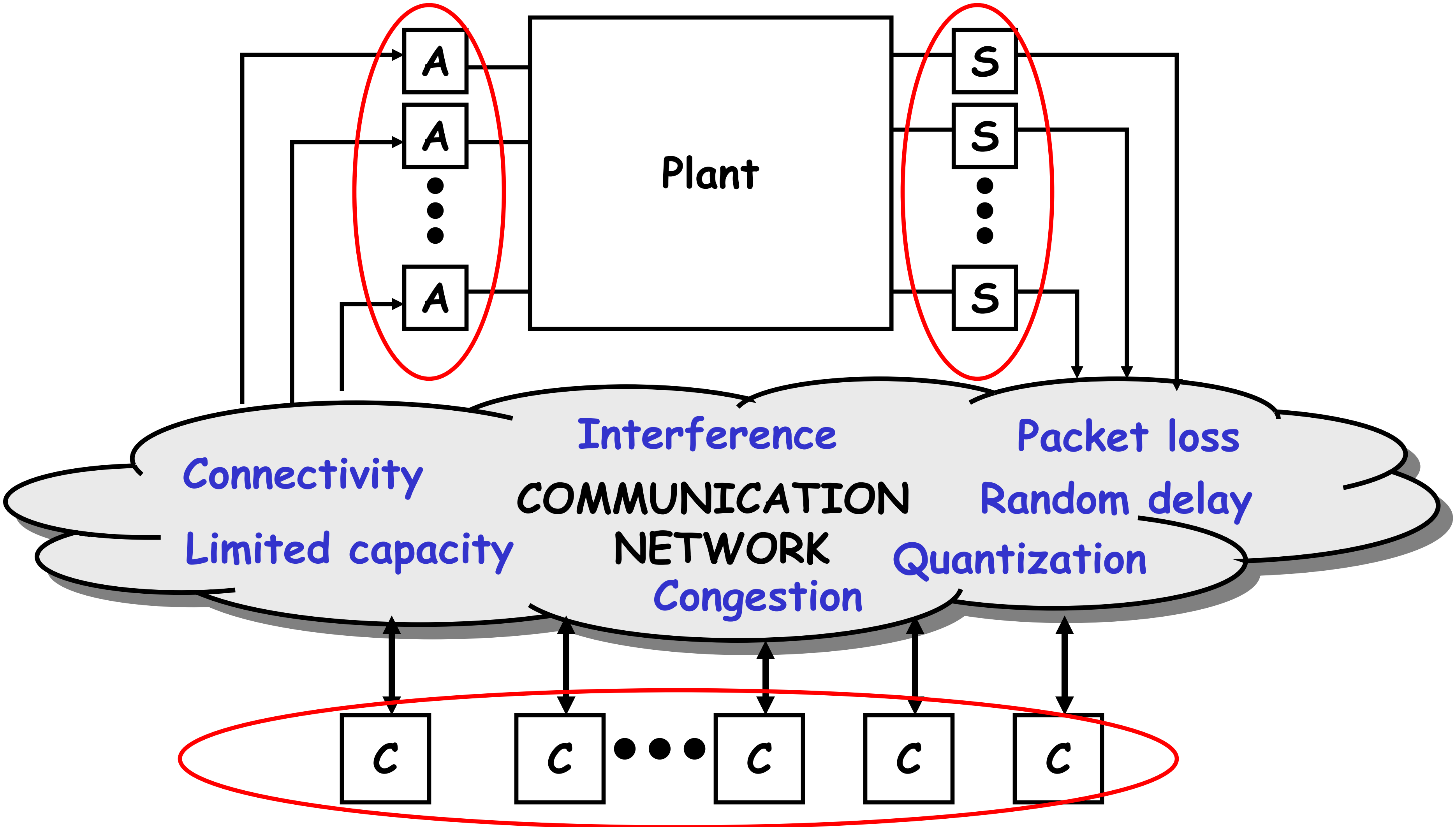
NCSs: what’s new for control?

Classical architecture: Centralized structure



NCSs: what's new for control?

NCSs: Large scale distributed structure



Outline

- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
  - Distributed Kalman filtering
- Open problems
  - Identification
  - Estimation
  - Control

The consensus problem

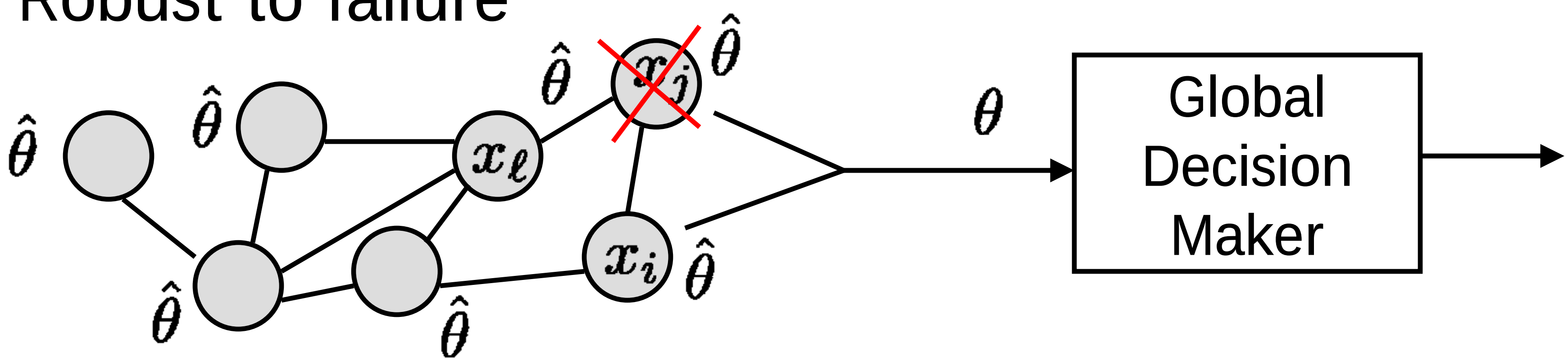
- Main idea
  - Having a set of agents to agree upon a certain value (usually global function) using only local information exchange (local interaction)
- Also known as:
  - Agreement problem (economics, signal processing, social networks)
  - Gossip algorithms (CS & communications)
  - Synchronization (statistical mechanics)
  - Rendezvous (robotics)
- Suitable for (noisy) sensor networks

Main features

- Distributed computation of general functions

$$\theta = f(x_1, \dots, x_N) = f\left(\frac{1}{N}\sum_{i=1}^{N} g_i(x_i)\right) \qquad \left(\text{ex. } \theta = \frac{1}{N}\sum_{i=1}^{N} x_i \text{ for } f = g = ident\right)$$

- Computational efficient (linear & asynchronous)
- Independent of graph topology
- Incremental (i.e. anytime)
- Robust to failure



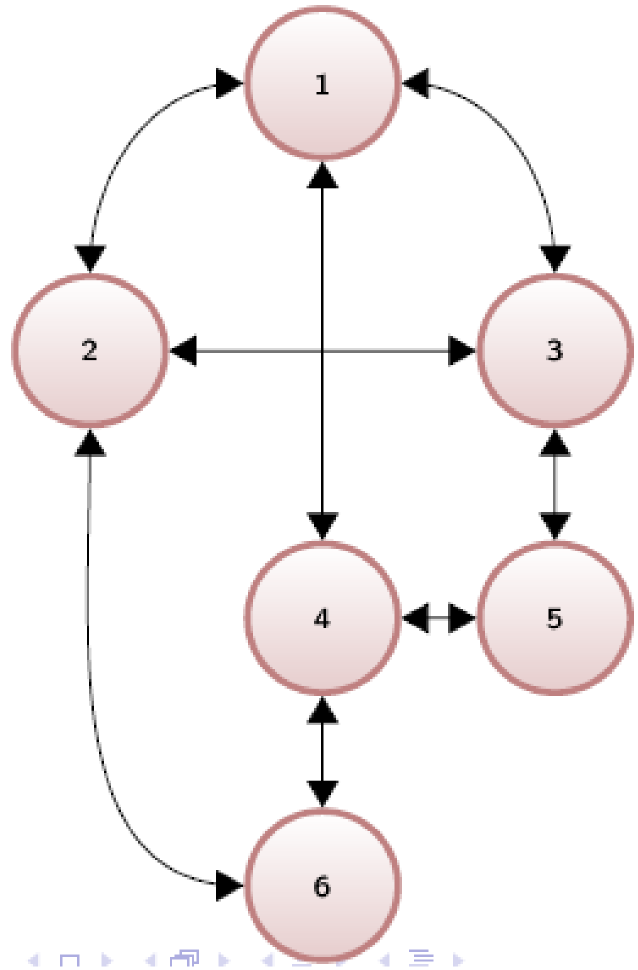
Some history (in control)

- Convergence of Markov Chains (60's) and Parallel Computation Alg.(70's)
- John Tsitsiklis "*Problems in Decentralized Decision Making and Computation* ", Ph.D thesis, MIT 1984
- A. Jadbabaie, J. Lin, and A. S. Morse "*Coordination of groups of mobile autonomous agents using nearest neighbor rules*", CDC' 02 (Axelby Best Paper Award TAC)
- Time-varying topologies (worst-case)
  - **L. Moreau**, "*Consensus seeking in multi-agent systems using dynamically changing interaction topologies*," IEEE, Transactions on Automatic Control, vol 50, No. 2, 2005
  - **M. Cao, A. S. Morse, and B. D. O. Anderson**. "*Reaching a Consensus in a Dynamically Changing Environment: A Graphical Approach*." SIAM Journal on Control and Optimization, Feb 2008
- Randomized topologies
  - **S. Boyd, A. Ghosh, B. Prabhakar, D. Shah** "*Randomized Gossip Algorithms*", TIT 2006
  - **F. Fagnani, S. Zampieri**, "*Randomized consensus algorithms over large scale networks*", JSAC 08
- Applications:
  - Vehicle coordination: Jadbabaie, Francis's group, Tanner, ...
  - Kalman Filtering: Olfati Saber-Murray, Alighanbari-How, Carli-Chiuso-Schenato-Zampieri
  - Generalized means: Giarre',Cortes
  - Time-synchronization: Solis-P.R. Kumar,Osvlado-Spagnolini, Carli-Chiuso-Schenato-Zampieri
  - WSN sensor calibration and parameter identification: Bolognani-DelFavero-Schenato-Varagnolo

Consensus formulation

Network of

- N agents
- Communication graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$
- i-th node neighbors: $\mathcal{N}(i)$
- Every node stores a variable: node i stores x_i.

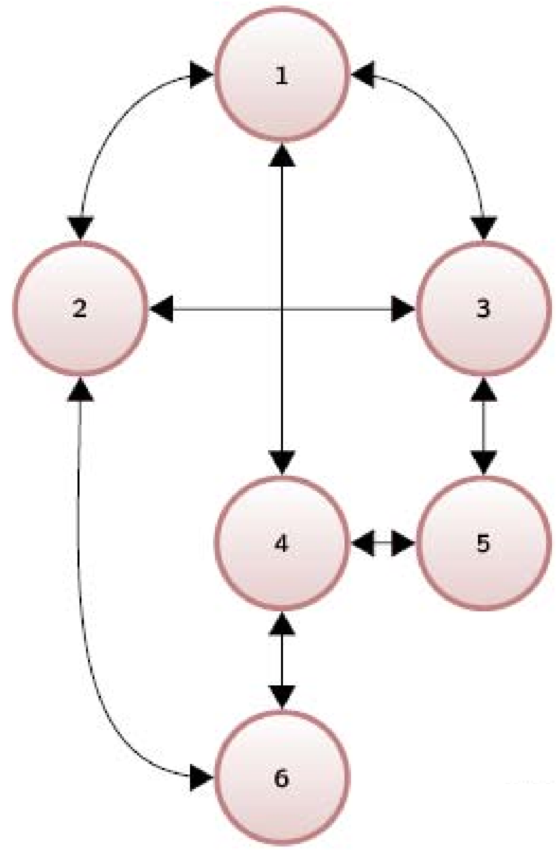


Consensus formulation (cont')

Definition (Recursive Distributed Algorithm adapted to the graph $\mathcal{G}$)

Any recursive algorithm where the i node's update law of depends only on the state of i and in its neighbors $j \in \mathcal{N}(i)$

$$x_i(t+1) = f(x_i(t), x_{j_1}(t), \dots x_{j_{N_i}}(t))$$
$$\text{with } j_1, \dots, j_{N_i} \in \mathcal{N}(i)$$

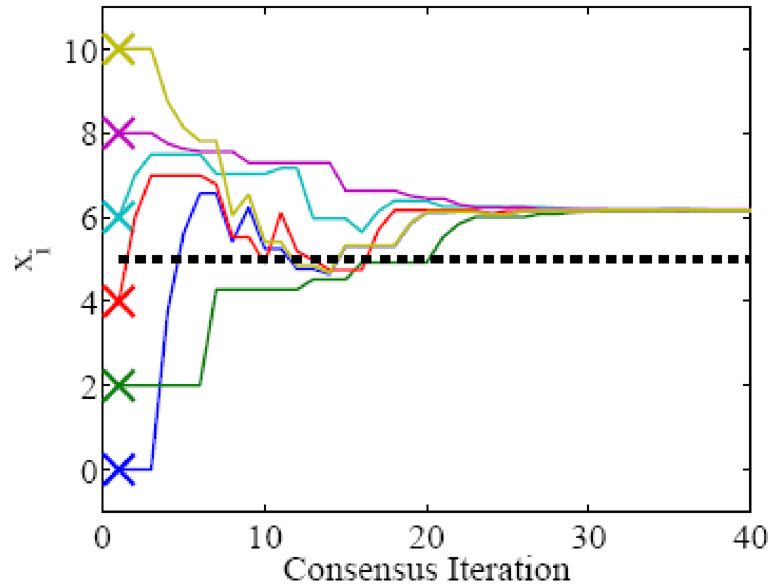


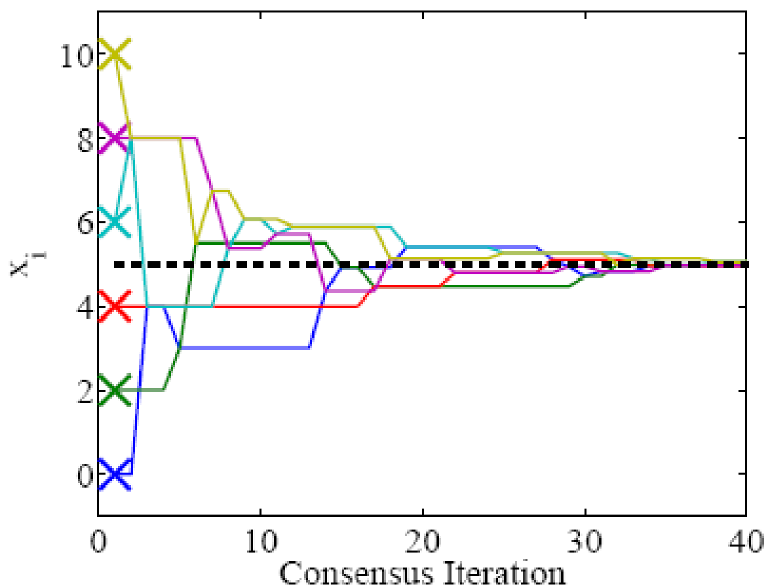
Consensus definition

Definition

A Recursive Distributed Algorithm adapted to the graph $\mathcal{G}$ is said to asymptotically achieve consensus if

$$x_i(t) \rightarrow \alpha \qquad \forall i \in \mathcal{N}$$





Definition

A Recursive Distributed Algorithm adapted to the graph $\mathcal{G}$ is said to asymptotically achieve *average* consensus if

$$x_i(t) \rightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} x_i(0) \qquad \forall i \in \mathcal{N}$$

Linear consensus

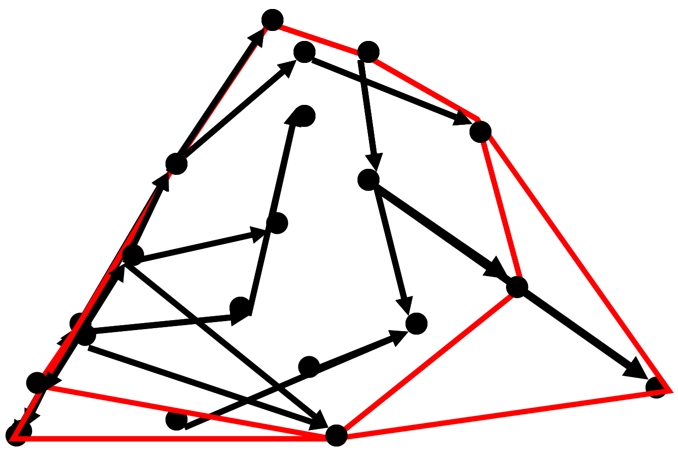
$$x_i(t+1) = p_{ii}x_i(t) + \sum_{j\in\mathcal{N}(i)} p_{ij}x_j(t)$$

$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_N(t) \end{bmatrix} \qquad x(t+1) = P(t)x(t)$$

Say $\mathcal{G}_P$ Graph associated to P, $P_{i,j} \neq 0 \iff (i,j) \in \mathcal{E}_P$,

$$\mathcal{G}_P \subseteq \mathcal{G} \qquad (\mathcal{N} \equiv \mathcal{N}_P, \quad \mathcal{E} \subseteq \mathcal{E}_P)$$

A robotics example: the rendezvous problem



$$x_i(t+1) = x_i(t) + u_i(t)$$
$$x_i(t+1) = p_{ii}x_i(t) + \sum_{j \in N(i)} p_{ij}x_j$$

Convex hull always shrinks.
If communication graph sufficiently connected, then shrinks to a point

Stochastic matrix

Definition (Stochastic Matrix)

If $P_{i,j} \geq 0$ and $\sum_j P_{i,j} = 1 \ \forall i$, than P is said to be stochastic

$$P\mathbb{1} = \mathbb{1} \qquad \mathbb{1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

Remark

If P is stochastic the linear algorithm can be written in both forms:

$$x_i(t+1) = p_{ii}x_i(t) + \sum_{j \in \mathcal{N}(i)} p_{ij}x_j(t)$$

$$x_i(t+1) = x_i(t) + \sum_{j \in \mathcal{N}(i)} p_{ij}\left(x_j(t) - x_i(t)\right)$$

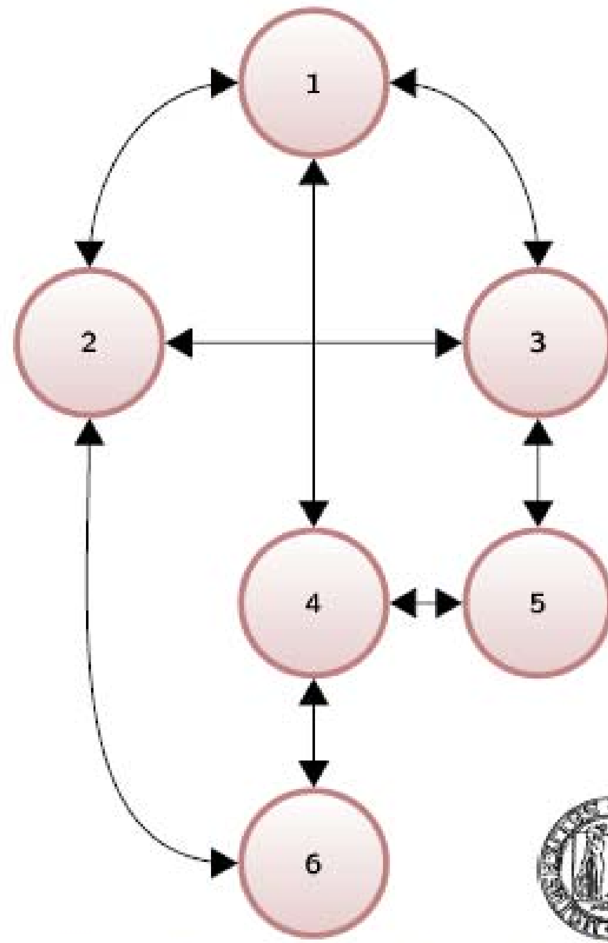
Constant matrix P

Synchronous Communication:

At each time all nodes communicate according to the communication graph

P(t)=P:

$$P=\begin{bmatrix} 3/6 & 1/6 & 1/6 & 1/6 & 0 & 0 \\ 1/6 & 3/6 & 1/6 & 0 & 0 & 1/6 \\ 1/6 & 1/6 & 3/6 & 0 & 1/6 & 0 \\ 1/6 & 0 & 0 & 3/6 & 1/6 & 1/6 \\ 0 & 0 & 1/6 & 1/6 & 4/6 & 0 \\ 0 & 1/6 & 0 & 1/6 & 0 & 4/6 \end{bmatrix}$$



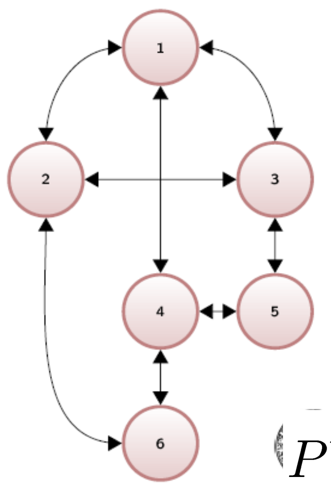
Convergence results

Theorem

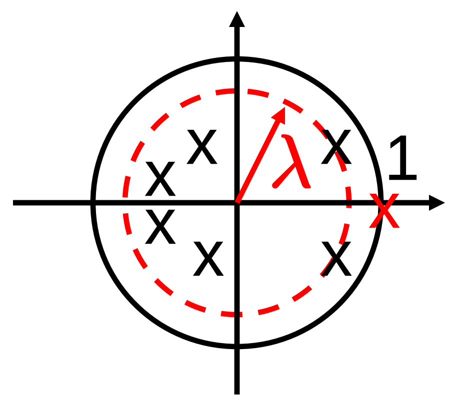
$P(t) = P$ stochastic.

- *If P such that $\mathcal{G}_P \subseteq \mathcal{G}$ is rooted then the algorithm achieves consensus*
- *If also P^T is stochastic (P doubly stochastic), then average consensus is achieved*

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} & p_{1,4} & 0 & 0 \\ p_{2,1} & p_{2,2} & p_{2,3} & 0 & 0 & p_{2,6} \\ p_{3,1} & p_{3,2} & p_{3,3} & 0 & p_{3,5} & 0 \\ p_{4,1} & 0 & 0 & p_{4,4} & p_{4,5} & p_{4,6} \\ 0 & 0 & p_{5,4} & p_{5,3} & p_{5,5} & 0 \\ 0 & p_{6,2} & 0 & p_{6,4} & 0 & p_{6,6} \end{bmatrix}$$



eigenvalues of P:



$P^t \xrightarrow{\lambda^t} \mathbb{1}\rho^T$, ρ is left eigenvector of 1

$\sum_i \rho_i = 1, \rho_i \geq 0, \quad (\rho_i > 0$ if strong. conn.)

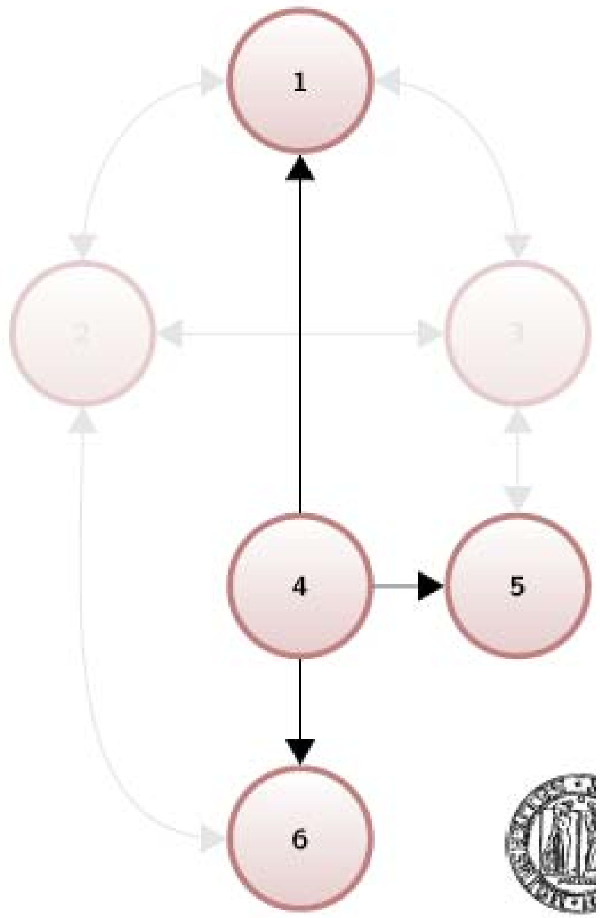
$\rho = \frac{1}{N}\mathbb{1}$ if P doubly stochastic

Time varying P(t): broadcast

Broadcast:

At each time one node randomly wakes up and broadcasts its information to all its neighbors.

$$P(t) = \begin{bmatrix} 3/4 & 0 & 0 & 1/4 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1/4 & 3/4 & 0 \\ 0 & 0 & 0 & 1/4 & 0 & 3/4 \end{bmatrix}$$

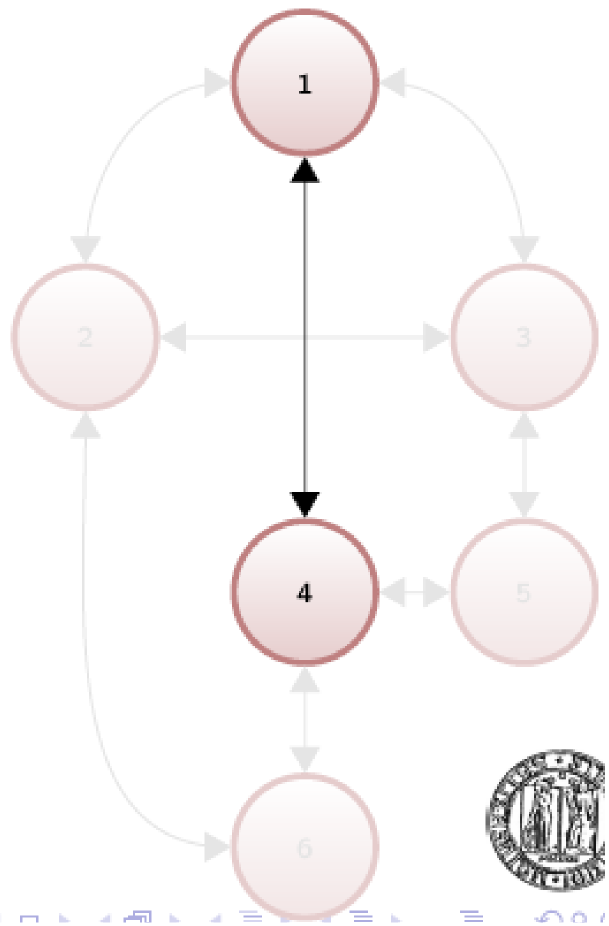


Time varying P(t): symmetric gossip

Symmetric Gossip:

At each time one node randomly wakes up and chose randomly a its neighbor. Those two nodes exchange information

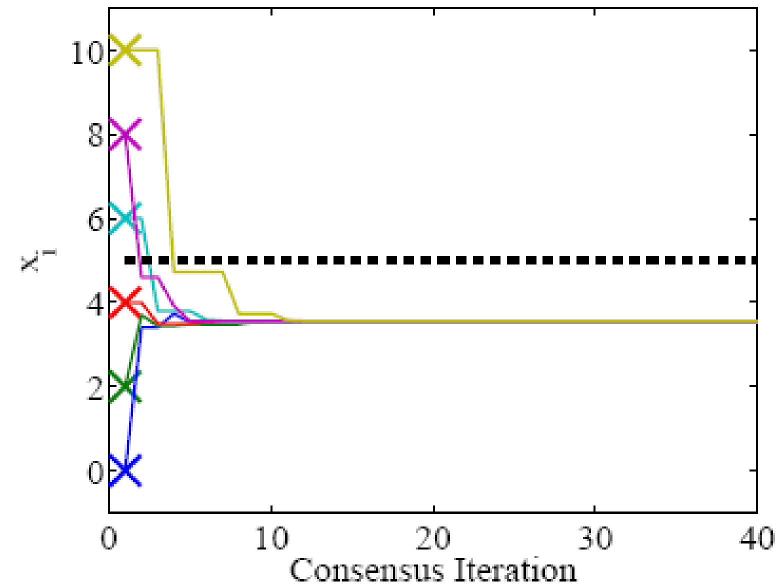
$$P(t) = \begin{bmatrix} 1/2 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

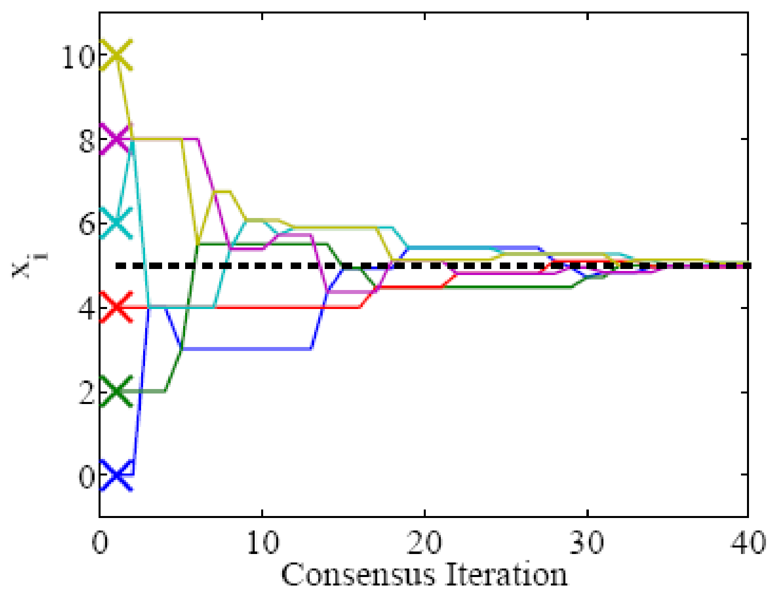


Consensus strategies for WSN

Broadcast

- 1 message broadcasted, $|\mathcal{N}(i)|$ estimate updated
- Does not guarantee average consensus





Symmetric Gossip

- At least 3 messages exchanged, 2 estimate updated
- Guarantee average consensus

Convergence results: P=P(t) deterministic

Theorem

Suppose that $P_{ii}(t) > 0, \forall i, \forall t$ and that there exixts K such that $\mathcal{G}_\ell = \mathcal{G}_{P((\ell+1)K)} \cup \ldots \cup \mathcal{G}_{P(\ell K)}$ is rooted at some node j for all ℓ then

- the sequence $\{P(t)\}$ achieves consensus
- if also $P^T(t)$ are stochastic for all t, then the sequence $\{P(t)\}$ achieves *average* consensus

Remark:

Estimates of rate of convergence are very conservative (worst case)

L. Moreau, "*Consensus seeking in multi-agent systems using dynamically changing interaction topologies*," IEEE, Transactions on Automatic Control, vol 50, No. 2, 2005
M. Cao, A. S. Morse, and B. D. O. Anderson. "*Reaching a Consensus in a Dynamically Changing Environment: A Graphical Approach*." SIAM Journal on Control and Optimization, Feb 2008

Convergence results: P=P(t) randomized

Theorem

Suppose $\{P(t)\}$ is a sequence of i.i.d. stochastic random matrices. Suppose moreover $\mathcal{G}_{P(t)} \subseteq \mathcal{G}$ $\forall t$ and call $\bar{P} = \mathbb{E}[P]$.

- *If $\mathcal{G}_{\bar{P}}$ is rooted that consensus is achieved w.p.1*
- *If also $P(t)^T$ is stochastic for every t, then average consensus is achieved w.p.1*

Remark:

It is not sufficient $\bar{P}$ doubly stochastic to guarantee average consensus

$$x(t+1) = P(t)x(t) = P(t)P(t-1)\cdots P(0)x(0) = Q(t)x(0) \qquad (Q(t) = P^t \text{ if } P(t) = P)$$

$$Q(t) \to \mathbb{1}\rho^T,\ \mathbb{E}[\rho] = \frac{1}{N}\mathbb{1},\ \boxed{Var(\rho) \sim \frac{1}{N}}$$

F. Fagnani, S. Zampieri, "*Randomized consensus algorithms over large scale networks*", IEEE Journal on Selected Areas in Communications, 2008

Generalized mean

$$x_i(t+1) = p_{ii}x_i(t) + \sum_{\mathcal{N}_i} p_{ij}x_j(t) \qquad x_i(t) \to \frac{1}{N}\sum_i x_i(0)$$

$$\theta = f(a_1, \ldots, a_N) = f\left(\frac{1}{N}\sum_{i=1}^N g_i(a_i)\right)$$

$$\boxed{\begin{aligned} x_i(0) &= g_i(a) \\ \hat{\theta}_i(t) &= f(x_i(t)) \end{aligned}}$$

$$\text{Geometric mean: } \theta = \sqrt[n]{\prod_i a_i} = \exp\left(\frac{1}{N}\sum_i \log(a_i)\right)$$

$$x_i(0) = \log(a_i), \quad \hat{\theta}_i(t) = \exp(x(t)) \to \theta$$

$$\text{Armonic mean: } \theta = \left(\frac{1}{N}\sum_i \frac{1}{a_i}\right)^{-1} \qquad x_i(0) = \frac{1}{a_i}, \quad \hat{\theta}_i(t) = \frac{1}{x_i(t)} \to \theta$$

$$\text{Quadratic mean: } \theta = \sqrt{\frac{1}{N}\sum_i a_i^2} \qquad x_i(0) = a_i^2, \quad \hat{\theta}_i(t) = \sqrt{x_i(t)} \to \theta$$

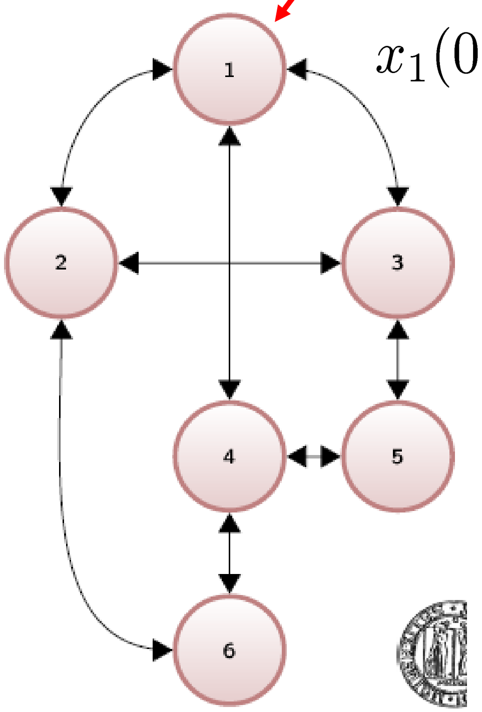
D. Bauso, L. Giarre' and R. Pesenti, "*Nonlinear protocols for Optimal Distributed Consensus in Networks of Dynamic Agents*", Systems and Control Letters, 2006

J. Cortés, *Distributed algorithms for reaching consensus on general functions,* Automatica 44 (3) (2008), 726-737

Node counting

$$x_i(t+1) = p_{ii}x_i(t) + \sum_{\mathcal{N}_i} p_{ij}x_j(t) \qquad x_i(t) \to \frac{1}{N}\sum_i x_i(0)$$

leader



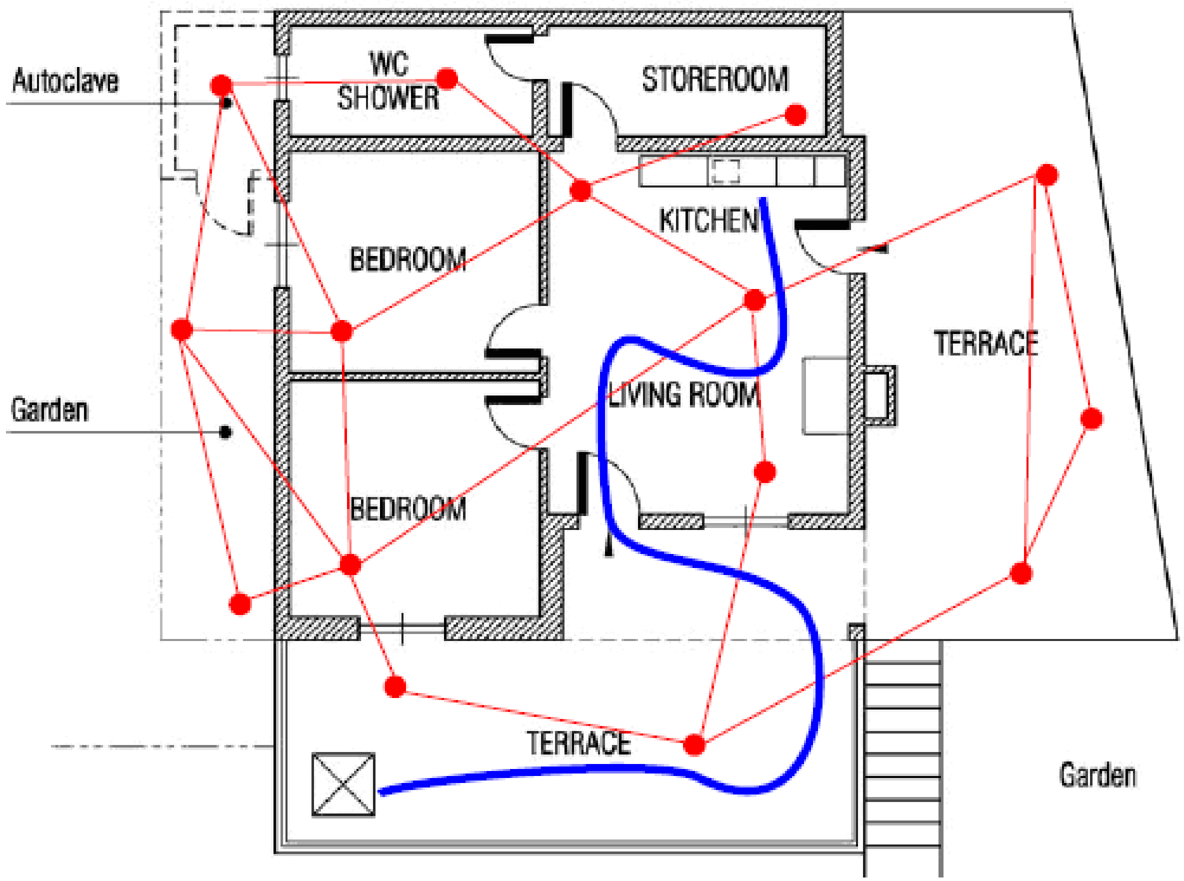
$$x_1(0) = 1, \quad x_i(0) = 0, i = 2, \ldots, N \implies \frac{1}{N}\sum_i x_i(0) = \frac{1}{N}$$

$$\boxed{\hat{\theta}_i(t) = \frac{1}{x_i(t)} \to N}$$

Outline

- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
  - Distributed Kalman filtering
- Open problems
  - Identification
  - Estimation
  - Control

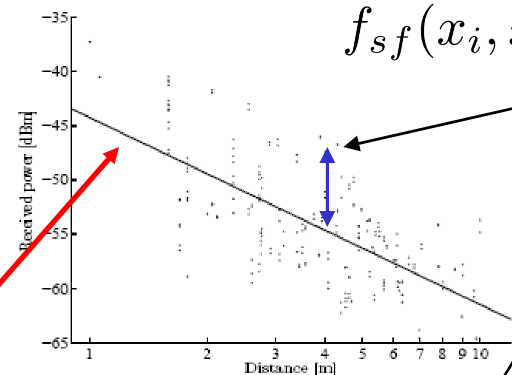
Localization with WSN



Localization with WSN

Each node

can measure the Radio Signal Strength Indicator, RSSI, i.e. the received signal power P_{rx} in dBm.



$$f_{sf}(x_i, x_j) = f_{sf}(x_j, x_i)$$

$$P_{rx}^{ij} = P_{tx}^{j} + \beta - 10\gamma \log_{10}(||x_i - x_j||) + f_{sf}(x_i, x_j) + v(t) + o_i$$

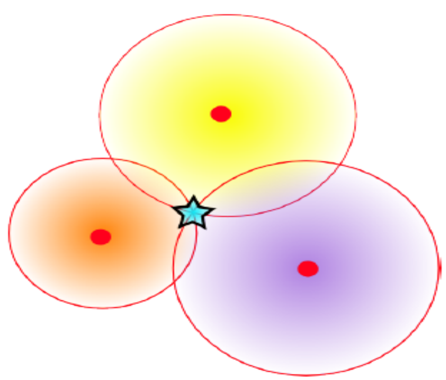
Transmi

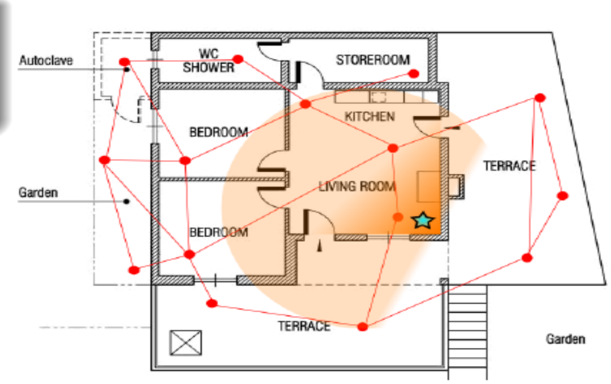
pow

Map Based

Most likely location that matches with pre-learned maps.

neter



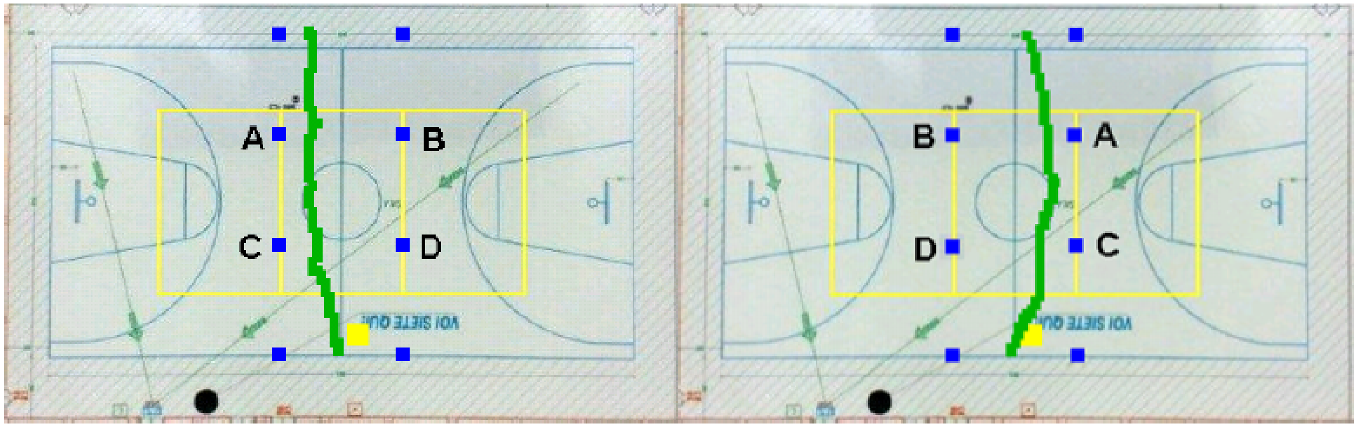


Range based

Triangulation (similarly to GPS)

Offset effect

Reception offset is particularly harmful for localization applications, Experiment inside a basketball court.[S 07][2]



[2][S 07] Courtesy of ST Microelectronics,
I. Solida, "Localization services for IEEE802.15.4/Zigbee devices. Mobile node tracking (in Italian)", Master Thesis, Department of information Engineering, University of Padua, 2007

WSN sensor calibration

Ideally:

- Estimate o_i: $\hat{o}_i$
- Use $\hat{o}_i$ to compensate the offset: $o_i - \hat{o}_i = 0$

Remember the previous example

What we propose is:

$$o_i - \hat{o}_i = \alpha \qquad \alpha \cong 0 \qquad \text{equal for all nodes}$$

All nodes overestimate or underestimate the distance similarly.
The errors, in the triangulation process, cancel out partially.

Calibration as consensus problem

Remark

If P is stochastic the linear algorithm can be written in both forms:

$$x_i(t+1) = p_{ii}x_i(t) + \sum_{j \in \mathcal{N}(i)} p_{ij}x_j(t)$$

$$x_i(t+1) = x_i(t) + \sum_{j \in \mathcal{N}(i)} p_{ij}\left(x_j(t) - x_i(t)\right)$$

$$o_i - \hat{o}_i(t) = x(t)$$

$$o_i - \hat{o}_i(t+1) = o_i(t) - \hat{o}_i(t) + \sum_{j \in \mathcal{N}_i} p_{ij}\left((o_i - \hat{o}_i(t)) - (o_j - \hat{o}_j(t))\right)$$

$$\boxed{\hat{o}_i(t+1) = \hat{o}_i(t) - \sum_{j \in \mathcal{N}_i} p_{ij}\left(P^{ij} - P^{ji} - \hat{o}_i(t) + \hat{o}_j(t)\right)}$$

update equation

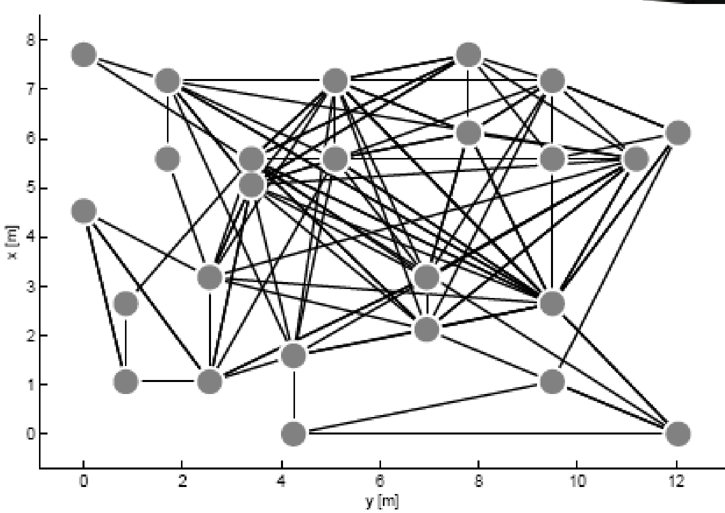
$$\boxed{\hat{o}_i(t) \to o_i - \frac{1}{N}\sum_i o_i = o_i - \alpha \approx o_i}$$

Steady state

Experimental Testbed

25 TMote-Sky nodes with Chipcon CC2420 RF Transceiver randomly placed inside a single conference room:

Network topology and nodes displacement:

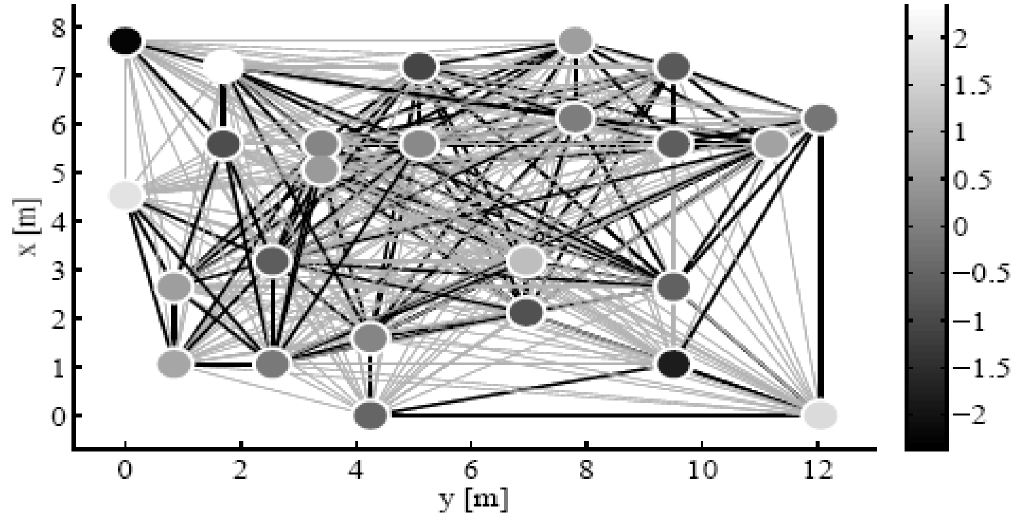


Kept just the links that safely carried the 75% of the sent messages over them

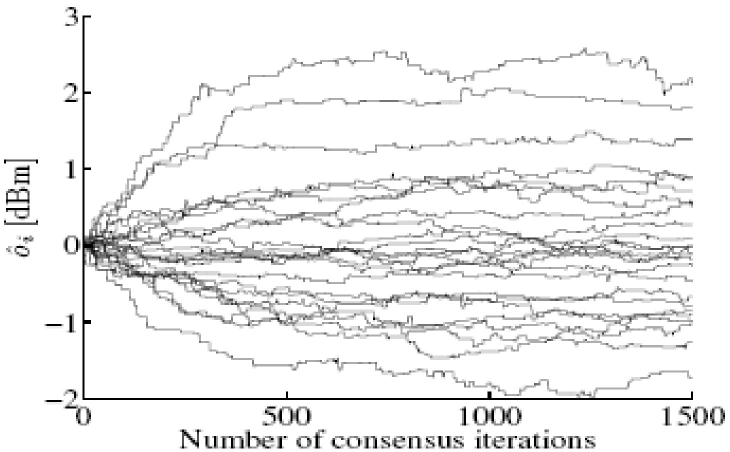
Experimental results

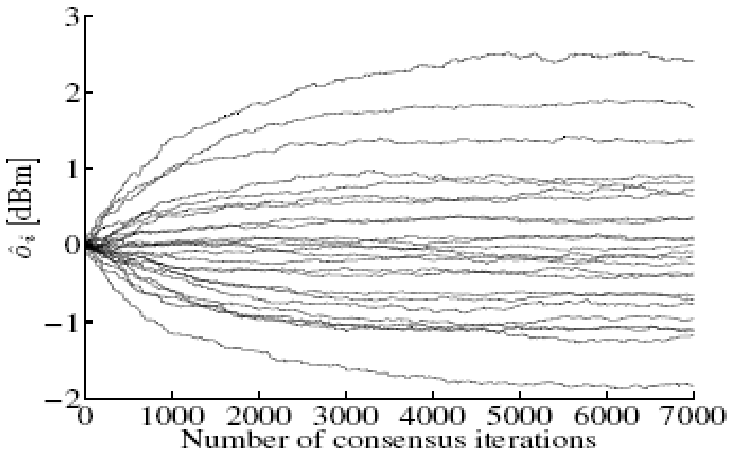
Links divided in 2 categories:

- Training links (black)
- Validation links (gray)

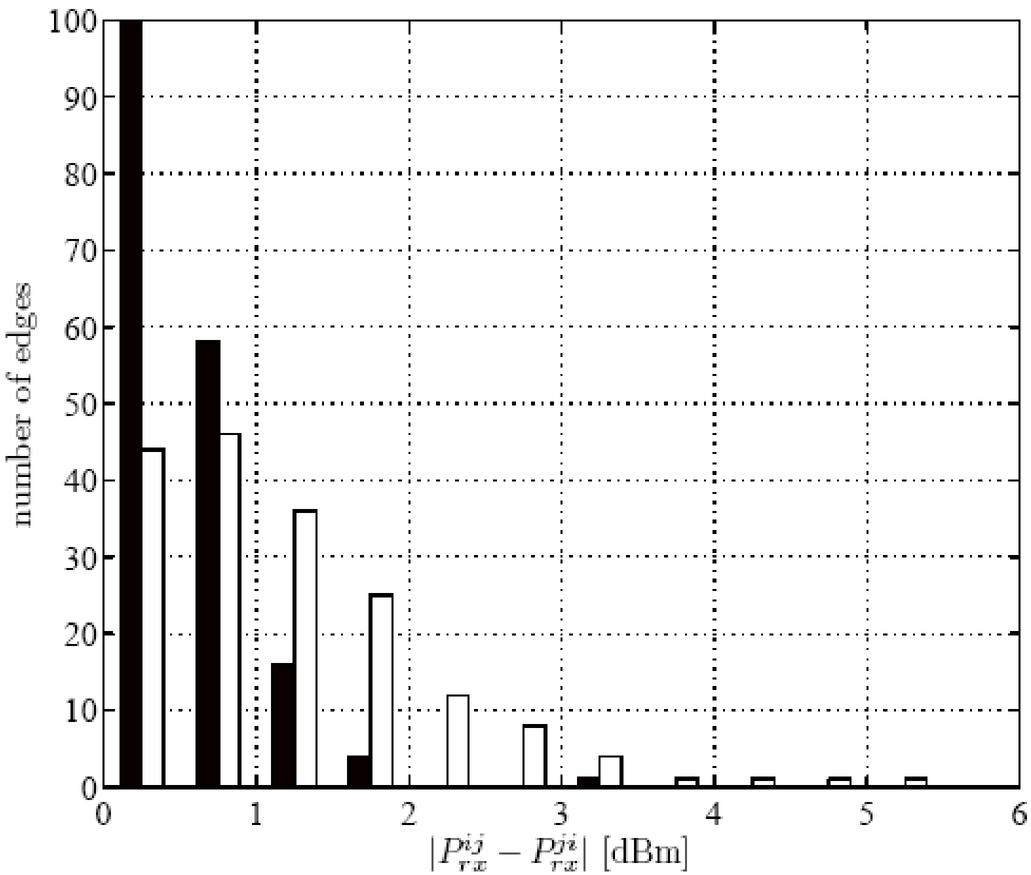


Estimate time evolution





$$\Delta \bar{P}^{ij} = \bar{P}^{ij} - \bar{P}^{ji} = o_i - o_j$$



	Before	After
<0.5 dB	24%	56 %
<1	50%	88 %
>2dB	35%	0.6 %
Max	<6dB	<3.5dB

Effects of systematic errors when estimating distances

$1dB \longmapsto \cong 2m \pm 0.28m.$
$6dB \longmapsto$ uncertainty for $0.9m$ to $4.4m$ for an actual distance of $2m$.

$1dB \longmapsto \cong 10m \pm 1.4m.$

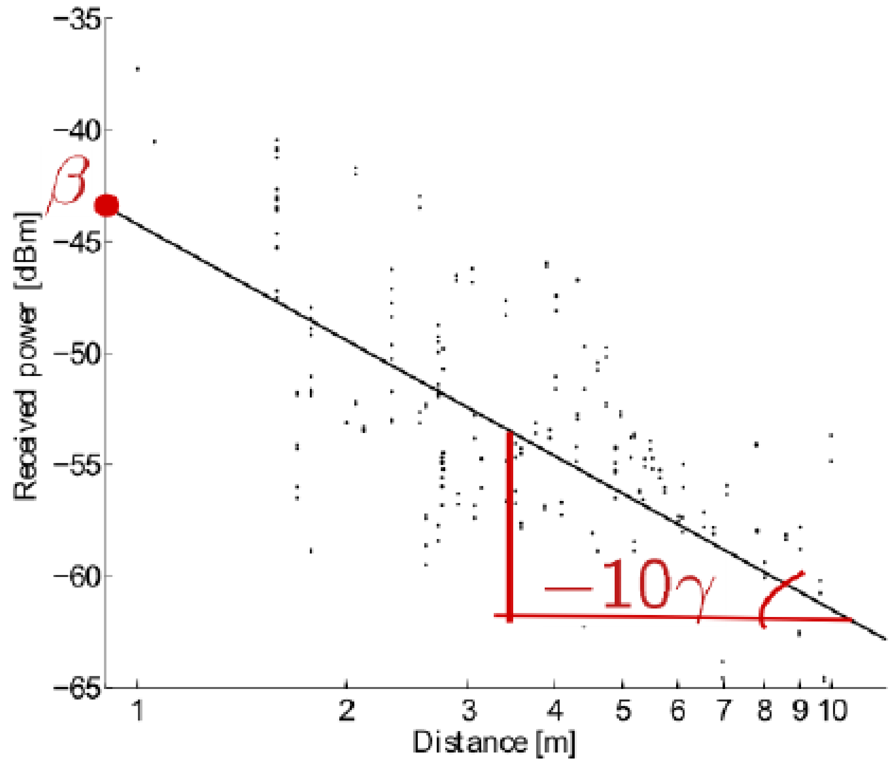
Parameter identification

Another important problem:

Accurately identify the wireless channel parameters β and γ.

In fact:

- Parameters extremely environment dependent
- $\gamma \in [1, 6]$
- Environment change hourly or daily



$$P_{rx}^{ij} = P_{tx}^{j} + \beta - 10\gamma \log_{10}(||x_i - x_j||) + f_{sf}(x_i, x_j) + v(t) + o_i$$

Modeling

Recall the Wireless Channel Model

$$\bar{P}_{rx}^{ij} + \hat{o}_i = P_{tx} - \beta - \gamma 10 \log_{10}(d_{ij}) + f_{sf}(\mathbf{x}_i, \mathbf{x}_j) + \underbrace{(o_i + \hat{o}_i)}_{\alpha} + w_i$$

$$\bar{P}_{rx}^{ij} + \hat{o}_i = \beta - \gamma 10 \log_{10}(d_{ij}) + w_i$$

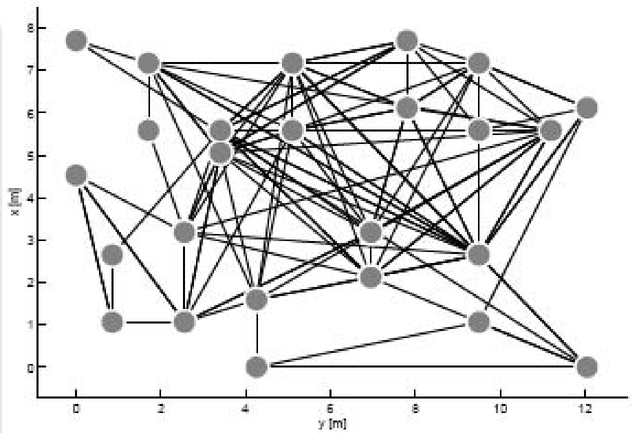
For each link:

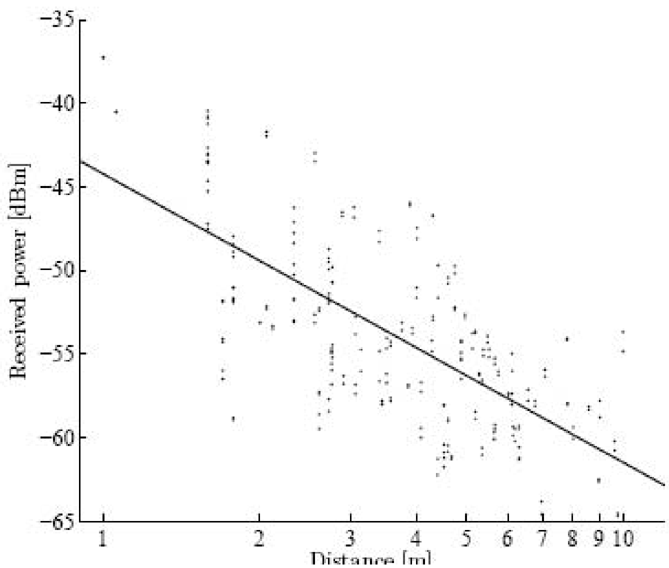
$$\boxed{\underbrace{\bar{P}_{rx}^{ij} + \hat{o}_i}_{b_{ij}} = \underbrace{[1 \quad -10 \log_{10}(d_{ij})]}_{a_{ij}^T} \underbrace{\begin{bmatrix} \beta \\ \gamma \end{bmatrix}}_{\theta} + w_i}$$

Modeling (cont'd)

Each node

- knows its distance with its neighbor
  $$d_{ij} \longrightarrow a_{ij}$$
- measures the strength of the message received form its neighbors
  $$P_{ij} \longrightarrow b_{ij}$$





Globally the network collected

M couples measure-regressors: $(a_1, b_1), \ldots (a_M, b_M)$

For ease of notation, assume that

Each node stores one couple measure-regressor.

Least-square Identification

Globally, the sensor network collected

M couples measure-regressors: $(a_1, b_1), \ldots (a_M, b_M)$.

Let us call

$$A = [a_1, \ldots, a_M]^T \text{ and } b = [b_1, \ldots, b_M].$$
$$b = A\theta + w$$

The least square estimate of θ,

given the measurements b is

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b$$

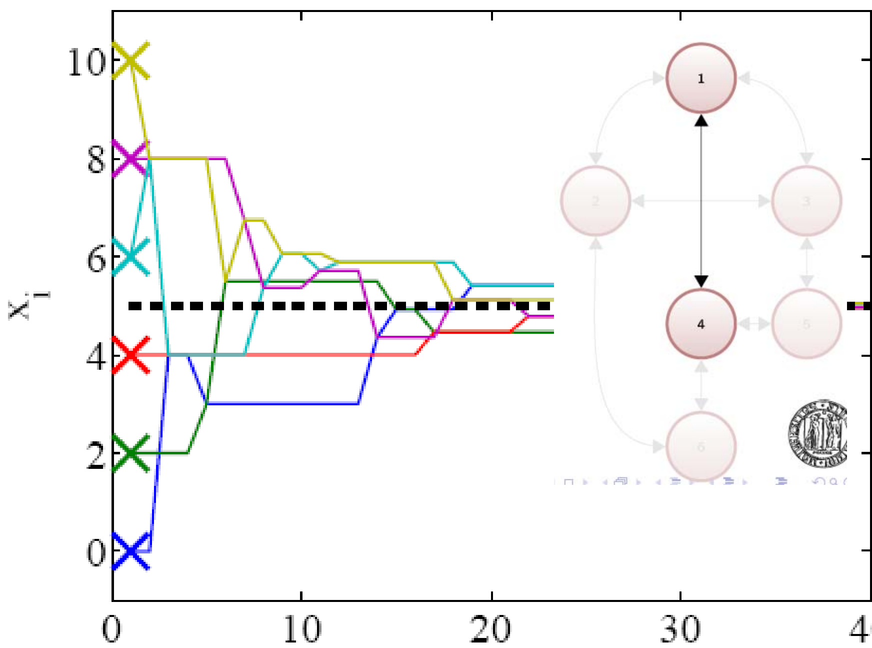
Consensus-based Identification

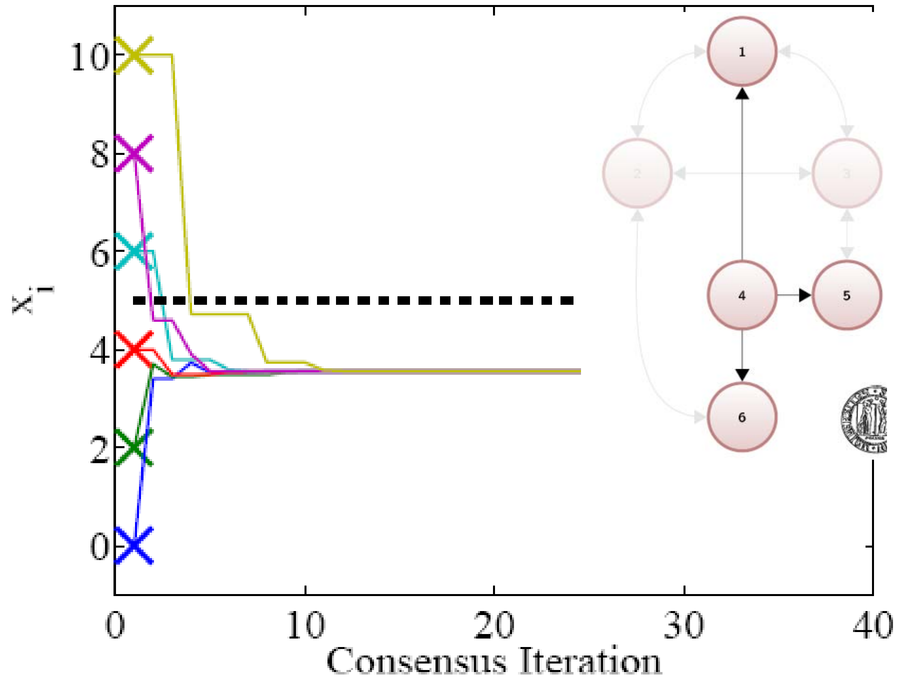
$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b = \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T\right)^{-1} \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i\right)$$

Communication schemes

Broadcast

- 1 message broadcasted, $|\mathcal{N}(i)|$ estimate updated
- Does not guarantee average consensus



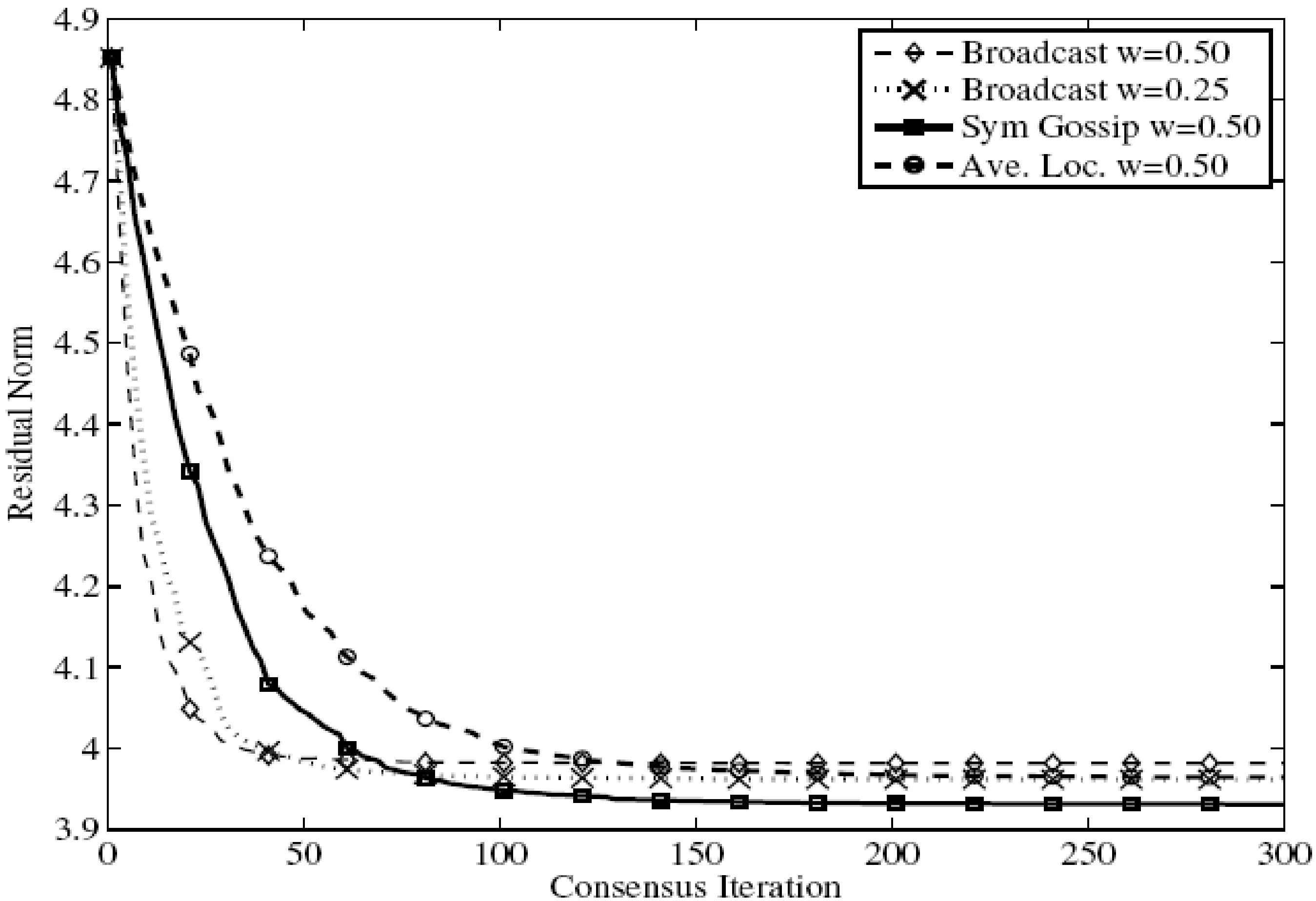


Symmetric Gossip

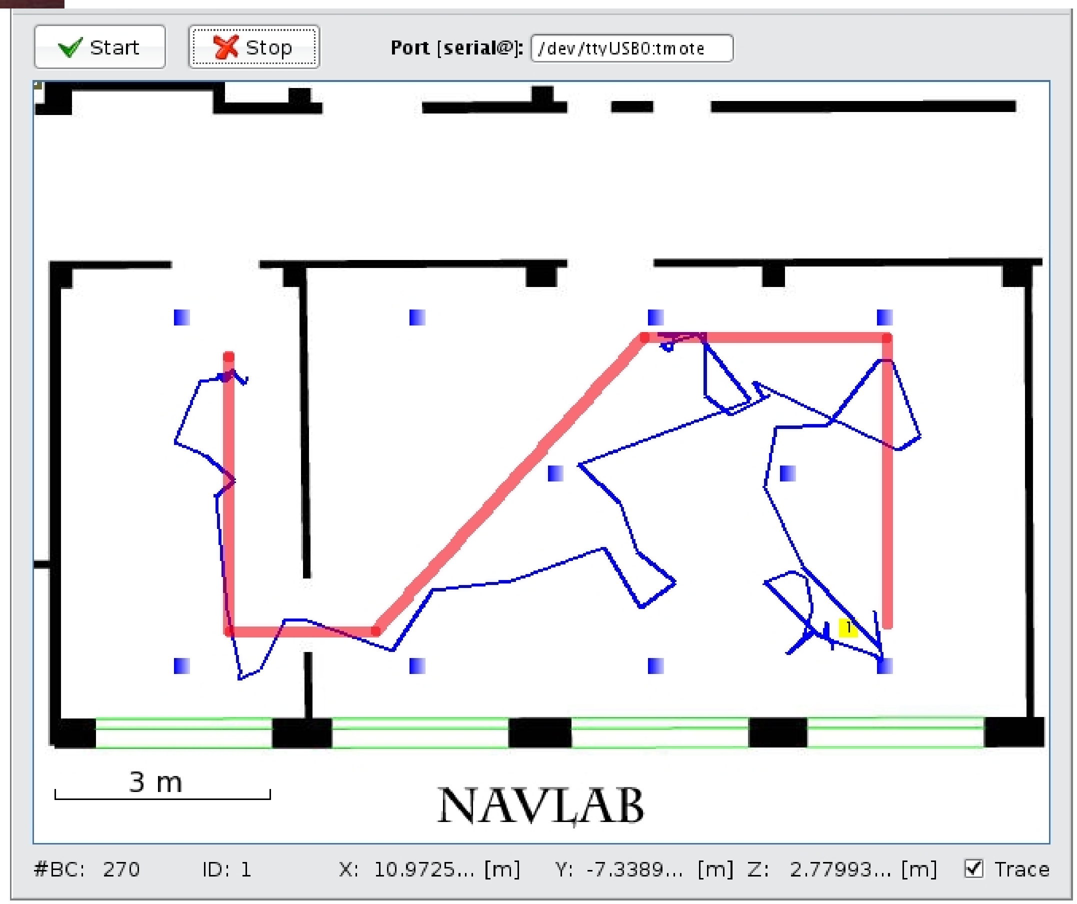
- At least 3 messages exchanged, 2 estimate updated
- Guarantee average consensus

Experimental results

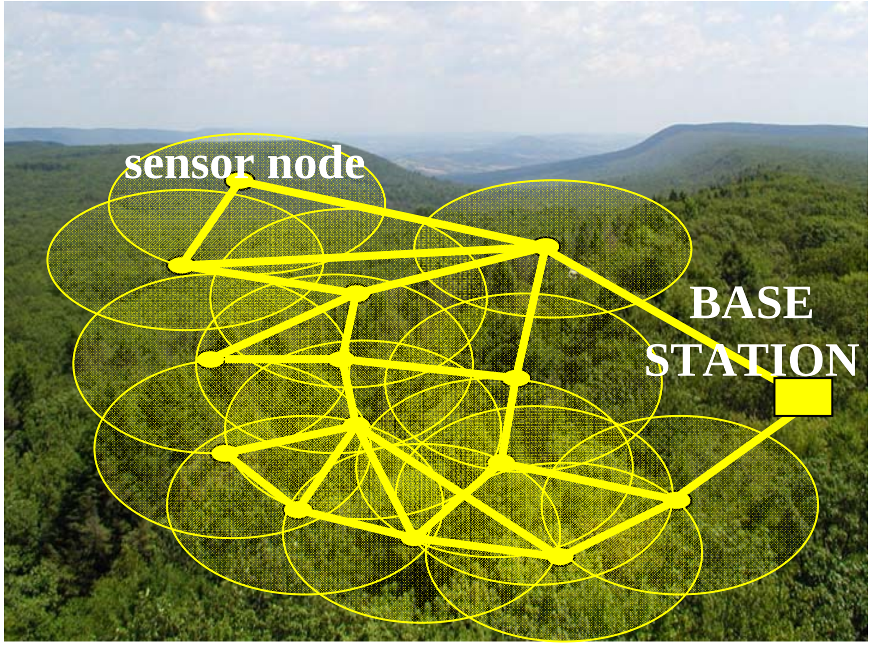
Residual: $\frac{1}{M}||A\hat{\theta} - b||^2$

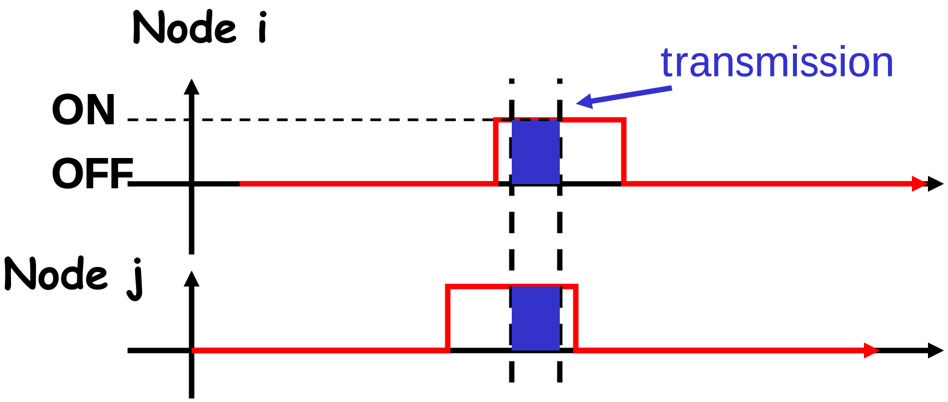


Tracking results

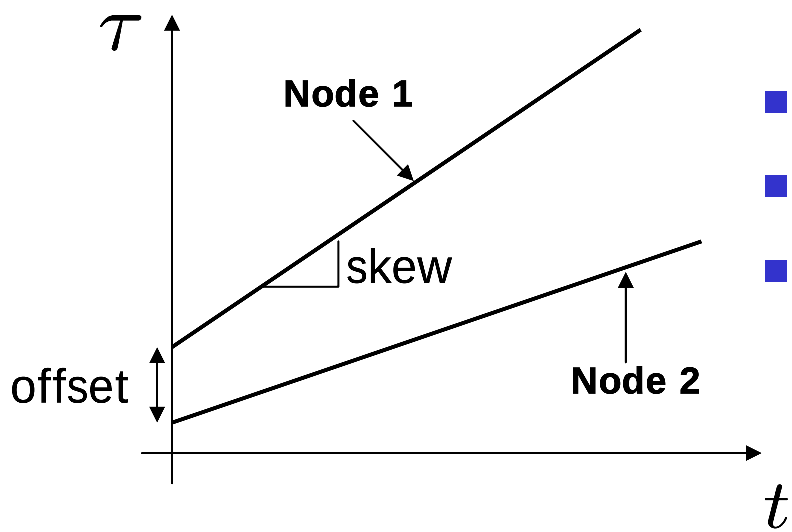


Time synchronization in sensor networks



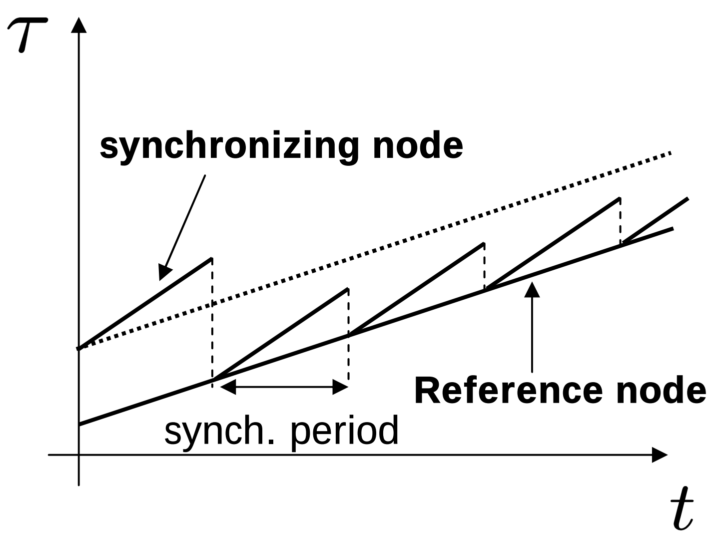


Clock characteristics & standard clock pair synch



- Offset: instantaneous time difference
- Skew: clock speed
- Drift: derivative of clock speed

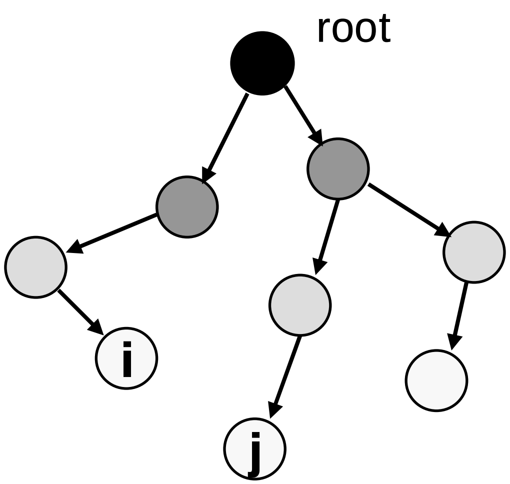
$$\tau_i = a_i t + b_i$$



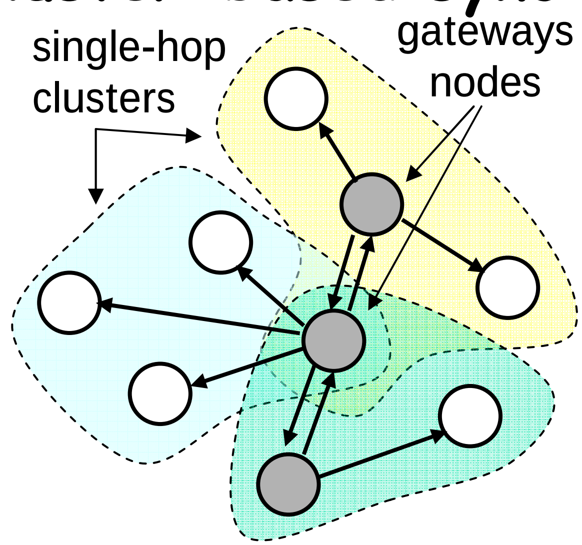
- Offset synch: periodically remove offset with respect to reference clock
- Skew compensation: estimate relative speed with respect to reference clock

State-of-the-art

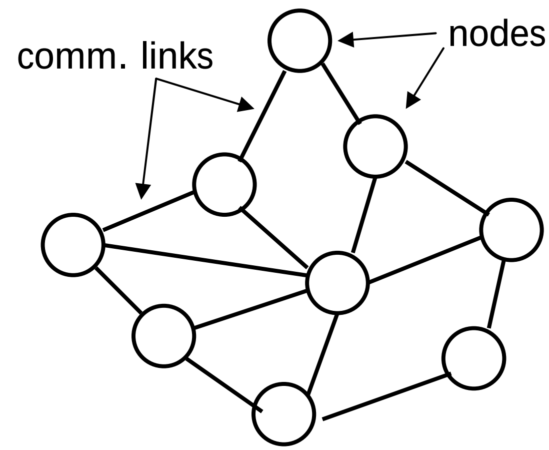
Tree-based sync



Cluster-based sync



Distributed



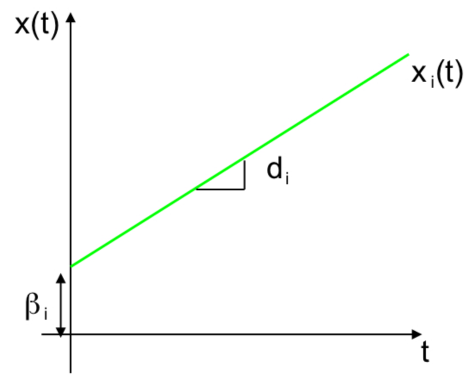
Modeling

MODEL: N clocks as discrete time integrators

$$\boxed{x_i(t+1) = x_i(t) + d_i}$$

d_i : *skew* (clock speed)

$x_i(0) = \beta_i$: *initial offset*



CONTROL: Assume that it is possible to control each clock by a local input $u_i(t)$:

$$x_i(t+1) = x_i(t) + d_i + u_i(t) \qquad \boxed{x(t+1) = x(t) + d + u(t)}$$

GOAL: Clocks Synchronization

$$\lim_{t\to\infty} x_i(t) - x_j(t) = 0 \qquad \boxed{\lim_{t\to\infty} y(t) = \lim_{t\to\infty} \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right) x(t) = 0}$$

CONTROL: Proportional controller

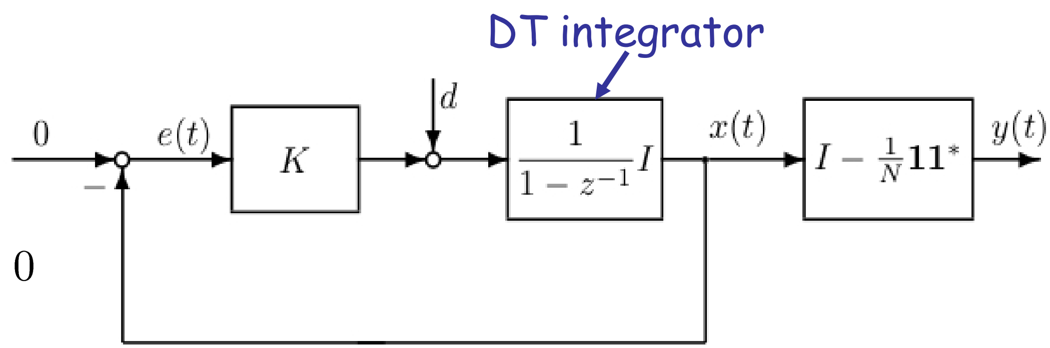
$$u_i(t) = -\sum_{j\in\mathcal{N}(i)} k_{ij}(x_j(t) - x_i(t)) \qquad \boxed{u(t) = -Kx(t)}$$

P-control

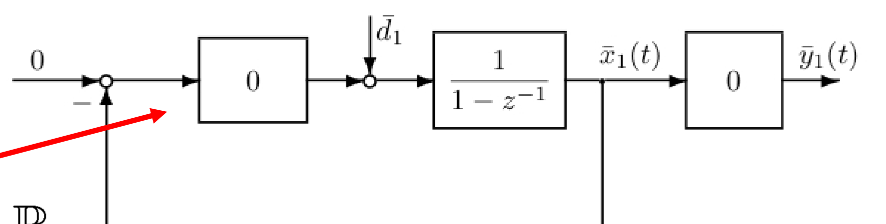
$$x(t+1) = x(t) + d + u(t)$$

$$u(t) = -Kx(t)$$

$$\lim_{t\to\infty} y(t) = \lim_{t\to\infty} \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right) x(t) = 0$$

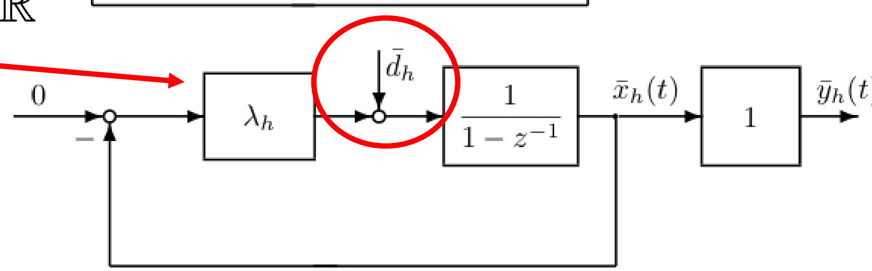


If K symmetric:



$\boxed{\mathbf{h = 1}}$

eingenvalues of $K \in \mathbb{R}$



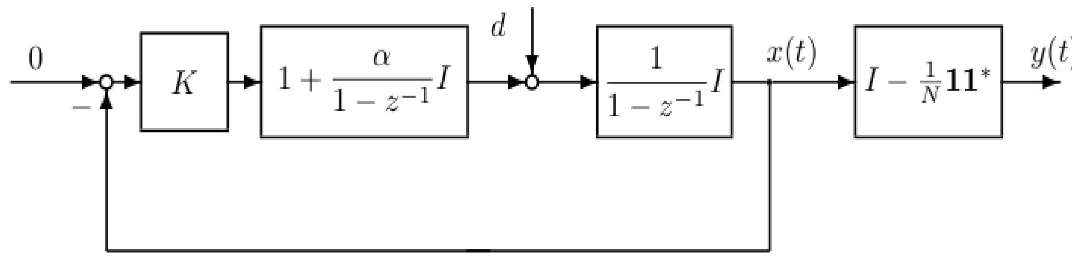
$\boxed{\mathbf{h = 2, \ldots, N}}$

PI-control

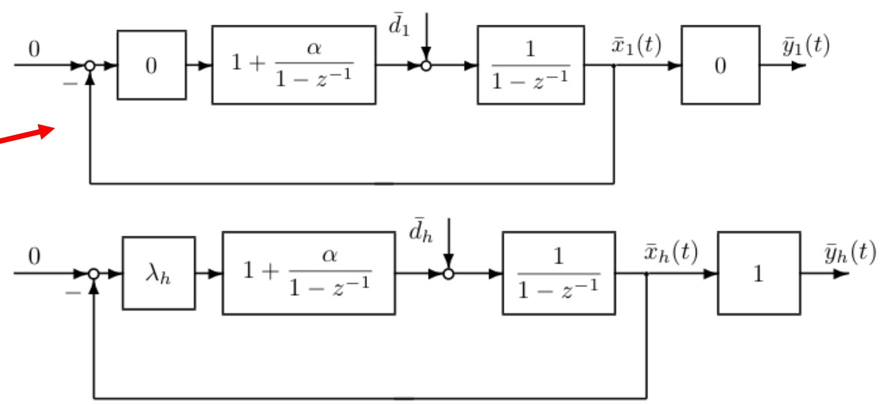
$$x(t+1) = x(t) + d + u(t)$$

$$\begin{aligned} w(t+1) &= w(t) - \alpha K x(t) \\ u(t) &= w(t) - K x(t) \end{aligned}$$

$$\lim_{t\to\infty} y(t) = \lim_{t\to\infty} \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T \right) x(t)$$



If K symmetric:



$\mathbf{h = 1}$

eingenvalues of $K \in \mathbb{R}$

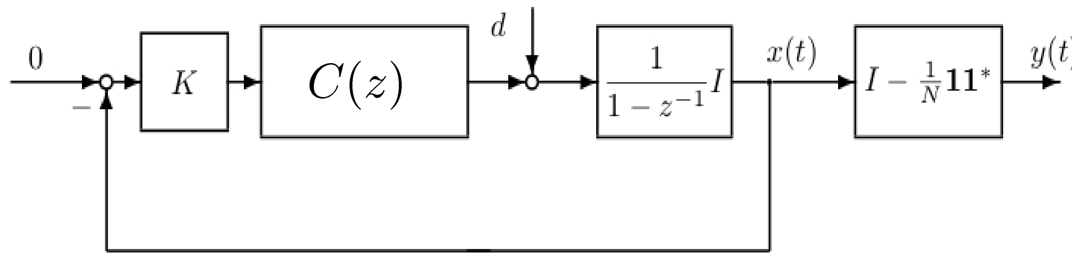
$\mathbf{h = 2, \ldots, N}$

C(z)-control

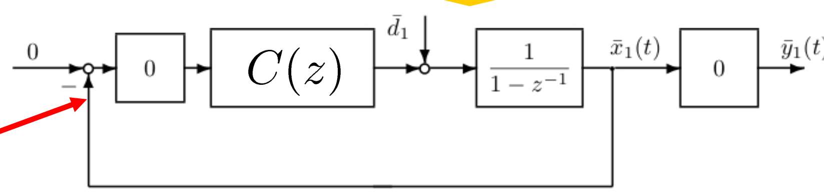
$$x(t+1) = x(t) + d + u(t)$$

$$u(t) \quad = \quad C(z)Kx(t)$$

$$\lim_{t\to\infty} y(t) = \lim_{t\to\infty} \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right) x(t)$$

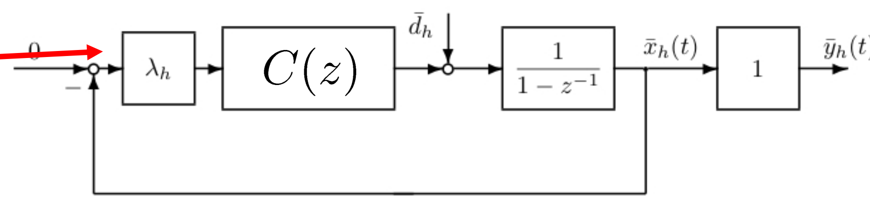


If K symmetric:



$$\mathbf{h = 1}$$

eingenvalues of $K \in \mathbb{R}$



$$\mathbf{h = 2, \ldots, N}$$

Parameter design (undirected graphs)

GOAL: fastest rate of convergence

$$\begin{bmatrix} x(t+1) \\ w(t+1) \end{bmatrix} = \begin{bmatrix} I - K & I \\ -\alpha K & I \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \end{bmatrix} + \begin{bmatrix} d \\ 0 \end{bmatrix}$$

- Suboptimal design (no topology needed):

$$k_{ij} = -\frac{1}{\max(d_i, d_j)+1} \; i \neq j, \quad \alpha = \frac{1}{2}, \quad \text{where } d_i \text{ is \# of neighbors of node } i.$$

- Optimal design: almost convex problem (SDP + 1D non-convex search)

Model w/ noise

$$u_i(t) = -\sum_{j \in \mathcal{N}(i)} k_{ij}(x_j(t) - x_i(t))$$

white measurement noise white process noise

$$\begin{bmatrix} x(t+1) \\ w(t+1) \end{bmatrix} = \begin{bmatrix} I-K & I \\ -\alpha K & I \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \end{bmatrix} + \begin{bmatrix} -K \\ -\alpha K \end{bmatrix} v(t) + \begin{bmatrix} d \\ 0 \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} n(t)$$

GOAL: smallest steady state mean square error: $J(K,\alpha) = \frac{1}{N} E[||y(\infty)||^2]$

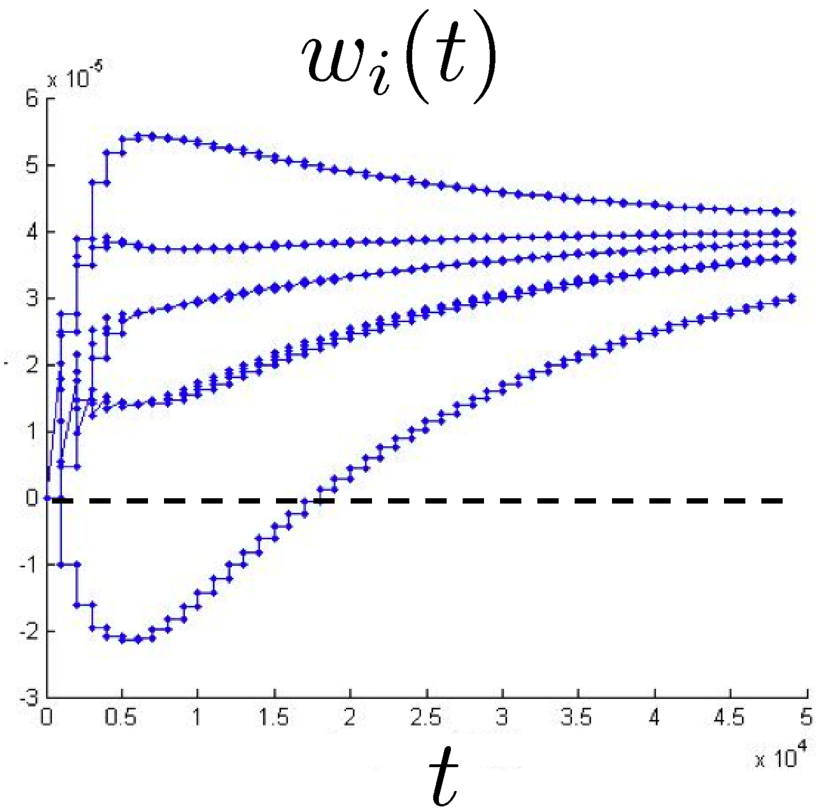
- Suboptimal design still OK

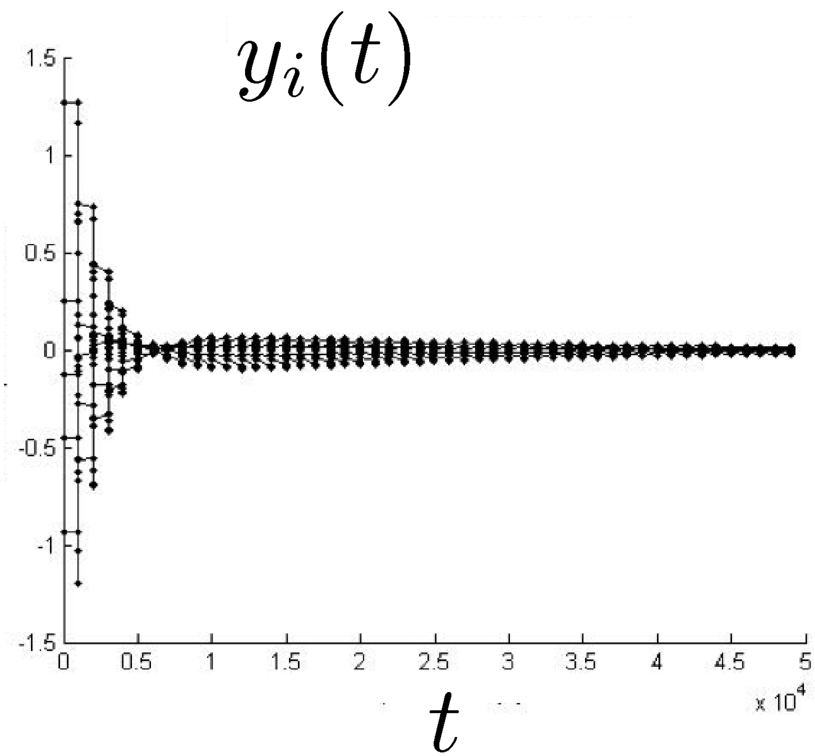
$$k_{ij} = \frac{1}{\max(d_i, d_j)+1}, \quad \alpha = \frac{1}{2}, \qquad \text{where } d_i \text{ is \# of neighbors of node } i.$$

- Optimal design: almost convex problem (Semidefinite programming in K+ 1D non-convex search in ff)

Simulations

Model parameters based on experimental data from real WSN and pseudo-synchronous implementation





$$y_i(\infty) = 0$$

$$w_i(\infty) + d_i = w_j(\infty) + d_j$$

Outline

- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
  - Distributed Kalman Filtering
- Open problems
  - Identification
  - Estimation
  - Control

Estimation framework

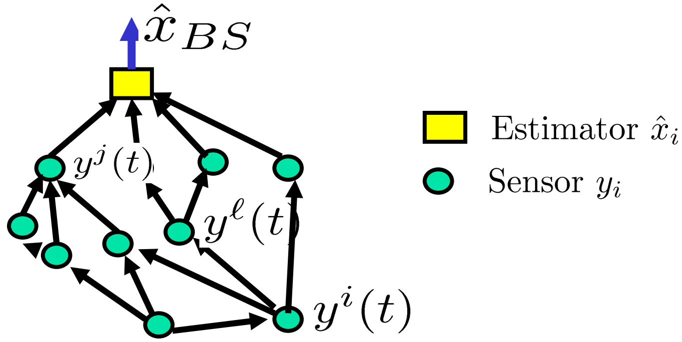
Static estimation

$$y(t) = C\theta + v(t)$$

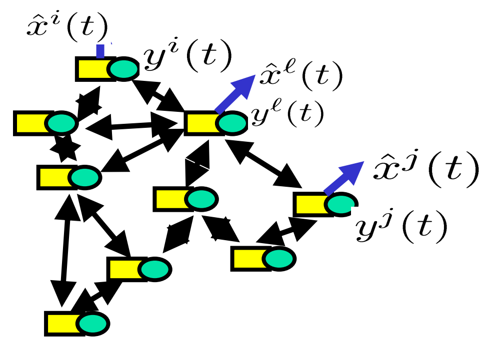
Dynamic estimation

$$\begin{aligned} x(t+1) &= Ax(t) + w(t) \\ y(t) &= Cx(t) + v(t) \end{aligned}$$

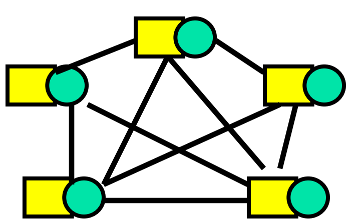
Hierarchical estimation



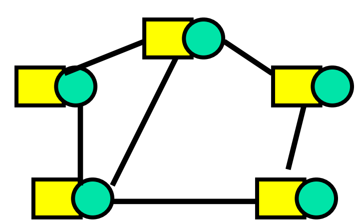
Distributed estimation



All-to-all communication



Multi-hop communication



Estimation framework

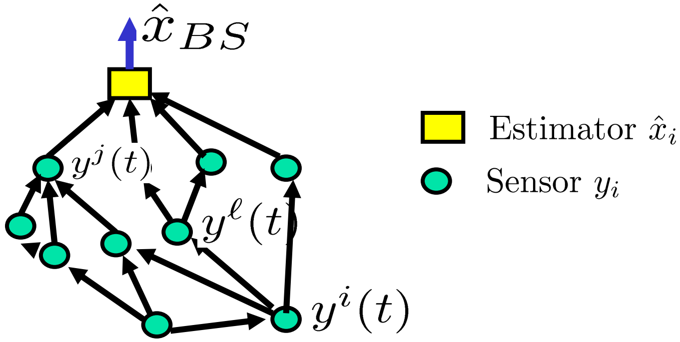
Static estimation

$$y(t) = C\theta + v(t)$$

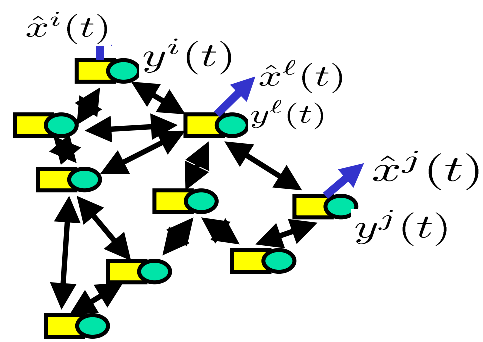
Dynamic estimation

$$\begin{aligned} x(t+1) &= Ax(t) + w(t) \\ y(t) &= Cx(t) + v(t) \end{aligned}$$

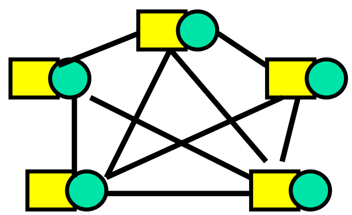
Hierarchical estimation



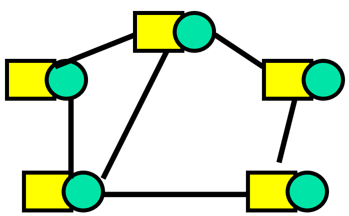
Distributed estimation



All-to-all communication



Multi-hop communication

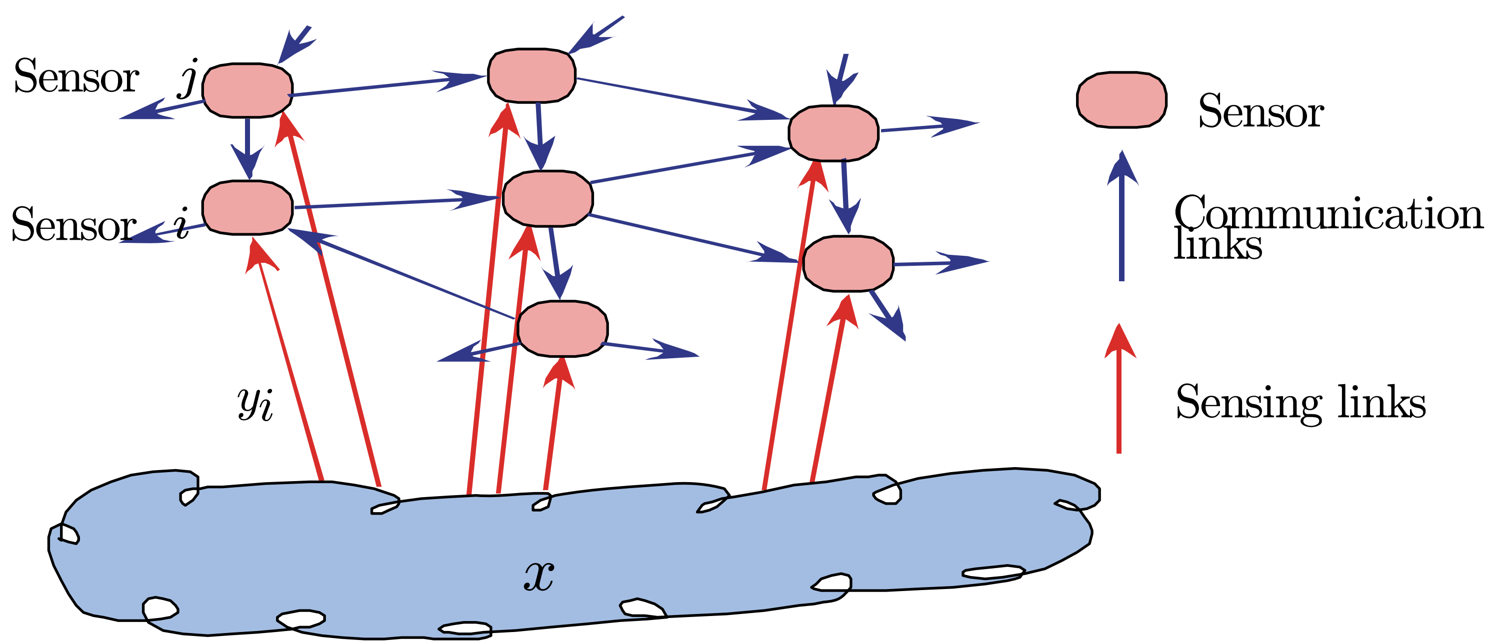


Problem setup

PROBLEM: N identical sensors measure a quantity $x \in \mathbf{R}$:

$$
\begin{aligned}
x(t+1) &= x(t) + w(t), \quad w(t) \sim \mathcal{N}(0,q) \\
y_i(t) &= x(t) + v_i(t), \quad v_i(t) \sim \mathcal{N}(0,r), \quad v_i \perp v_j, \quad i = 1,\dots,N
\end{aligned}
$$

Communication topology contrained to be consistent with communication graph:



Desired solution: centralized Kalman filter

Optimal estimator $\hat{x}(t|t) = \mathcal{E}[x(t)|y_1(0),..y_1(t),..,y_N(0),..,y_N(t)]$ with no graph constraint:

$$\hat{x}(t|t) \quad = (1-\ell_c)\hat{x}(t-1|t-1) + \ell_c \mathbf{mean}(y_i(t))$$

where ℓ_c is centralized Kalman gain and $\mathbf{mean}(y_i) = \frac{1}{N}\sum_i y_i$.

Decentralized solution with all-to-all communication:

$$\begin{aligned} \hat{x}^i(t|t) &= (1-\ell_c)\hat{x}^i(t-1|t-1) + \ell_c \mathbf{mean}(y_i(t)) \\ &= \mathbf{mean}\big((1-\ell_c)\hat{x}^i(t-1|t-1) + \ell_c y_i(t)\big) \end{aligned}$$

GOAL: Find algorithm to compute $\mathbf{mean}()$ in distributed fashion for multi-hop.

(POSSIBLE) SOLUTION: Linear average consensus algorithm: let Q doubly stochastic matrix compatible with communication graph $\mathcal{G}$, i.e. $Q_{ij} = 0$ if link $(i,j) \notin \mathcal{G}$

$$\begin{aligned} \hat{x}^i(t|t) &= \mathbf{mean}\big((1-\ell_c)\hat{x}^i(t-1|t-1) + \ell_c y_i(t)\big) \\ &\approx {\color{red}Q^m}\big((1-\ell_c)\hat{x}^i(t-1|t-1) + \ell_c y_i(t)\big), \quad {\color{red}m \gg 1} \end{aligned}$$

Distributed Kalman Filter

[Olfati-Saber,Spanos,Murray,Alriksson,Rantzer]

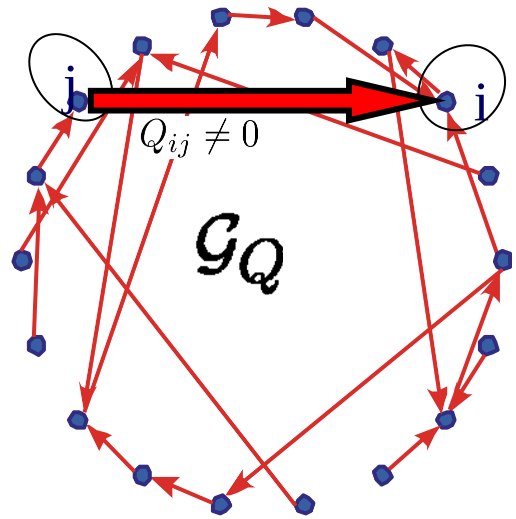
The i-th sensor build its estimate of $x(t)$ by a two-stage strategy.

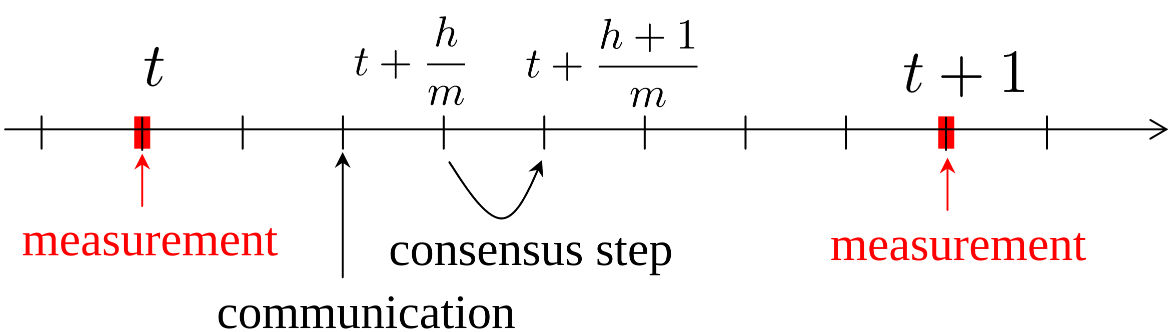
FIRST STEP: *Measurement Stage* $(0 < \ell < 1)$

$$\hat{x}_i(t|t) = (1-\ell)\hat{x}_i(t|t-1) + \ell y_i(t)$$

SECOND STEP: *Consensus Stage*

$$\hat{x}_i(t+1|t) = \sum_{j=1}^{N} Q_{ij}^{m} \hat{x}_j(t|t)$$





Steady-state performance

Let $\tilde{x}(t) := x(t)\mathbb{1} - \hat{x}(t)$ and $P(t) := \mathbb{E}[\tilde{x}(t)\tilde{x}^*(t)]$. The covariance matrix $P(t)$ satisfies:

$$\begin{aligned}\lim_{t\to\infty} P(t) &= r\ell^2 \sum_{i=0}^{\infty} (1-\ell)^{2i} Q^{(i+1)m} \left(Q^*\right)^{(i+1)m} \\ &+ q \frac{1}{1-(1-\ell)^2} \mathbb{1}\mathbb{1}^*\end{aligned}$$

$$J(\underbrace{Q, \ell}_{\text{optimization variables}}; \overbrace{m, r, q}^{\text{parameters}}) := \operatorname{trace}\left(\lim_{t\to\infty} P(t)\right)$$

COST FUNCTION

Objective

PROBLEM: Given a graph $\mathcal{G}$ and a nonnegative integer m, find $\ell \in (0,1)$ and a stochastic matrix Q in the set of stochastic matrices *compatible* with the graph $\mathcal{G}$, minimizing J, i.e.

$$(Q^{opt}, \ell^{opt}) \in \operatorname*{argmin}_{Q,\ \ell} J(Q, \ell, m, q, r)$$

REMARK: Can be generalized to quasi-stochastic matrices (i.e. Q_{ij} can be negative)

Assumption: Q normal

Consider the set of eigenvalues of Q

$$\sigma(Q) = \{1, \lambda_1, \lambda_2, \ldots, \lambda_{N-1}\}$$

Q is stochastic implies $\lambda_0 = 1$. If Q is a normal matrix ($QQ^* = Q^*Q$), then

$$\boxed{J = \frac{r\ell^2 + qN}{1-(1-\ell)^2} + r\ell^2 \sum_{i=1}^{N-1} \frac{|\lambda_i|^{2m}}{1-(1-\ell)^2|\lambda_i|^{2m}}}$$

ASSUMPTION: Q is normal (symmetric matrices, circulant matrices, Abelian Cayley matrices....).

REMARK: If Q normal, then $Q_{sym} = \frac{Q+Q^*}{2}$ is such that $J(Q_{sym}, \ell) \leq J(Q, \ell)$.

Convexity for fixed l or Q

THEOREM: The function $J(Q,\ell)$ is a convex function in $\mathcal{Q}_{sym}$.

$+$

$\mathcal{Q}_{sym}$ is a convex set.

$$Q^{opt}(\ell) = \underset{Q \in \mathcal{Q}_{sym}}{\operatorname{argmin}} \; J(Q,\ell,m,q,r) \textit{ is a convex problem for fixed } \ell.$$

The solution can be performed by efficient numerical tools (Boyd, Xiao...)

$$\ell^{opt}(Q) = \underset{\ell}{\operatorname{argmin}} \, J(Q,\ell,m,q,r) \textit{ is a convex problem for fixed } Q.$$

Joint optimization: special cases

Unfortunately J is **NOT** a convex function jointly in ℓ and Q.

However, an analytical characterization is possible when restricting to some asymptotic cases on the values of m, r and q. In particular:

- fast communication, i.e., $m \to \infty$
- $\frac{r}{q} \approx 0$, i.e. small measurement noise
- $\frac{q}{r} \approx 0$, i.e. small process noise

Fast communication

FAST COMMUNICATION: $m \to \infty$

$\rho(Q)$: *the essential spectral radious* of the matrix Q, namely the second largest eigenvalue in absolute value.

$$(Q^{opt}(m), \ell^{opt}(m)) = \underset{Q,\ \ell}{\operatorname{argmin}}\, J(Q, \ell; m, r, q)$$

Theorem

Let

$$\bar{Q} = \underset{Q}{\operatorname{argmin}}\, \rho(Q), \quad \bar{Q} \text{ unique.}$$

Then

$$\lim_{m \to \infty} \ell^{opt}(m) = \ell_c^{opt}.$$

and

$$\lim_{m \to \infty} Q^{opt}(m) = \bar{Q}.$$

Small measurement noise

SMALL MEASUREMENT NOISE: $r/q \;\to\; 0$

Frobenious norm: $\;||Q||_F := (\text{trace}\{QQ^*\})^{1/2}$

$$(Q^{opt}(r/q), \ell^{opt}(r/q)) = \operatorname*{argmin}_{Q,\;\ell} J(Q,\ell;m,r,q)$$

Theorem

Let

$$\bar{Q} = \operatorname*{argmin}_{Q} \|Q^m\|_F, , \quad \bar{Q} \text{ unique.}$$

Then

$$\lim_{r/q\to 0} Q^{opt}(r/q) = \bar{Q}.$$

and

$$\ell^{opt}(r/q) = 1 - \frac{\|\bar{Q}\|_F^2}{N}\frac{r}{q} + o\,(r/q)\,.$$

High measurement noise

HIGH MEASUREMENT NOISE: $q/r \;\rightarrow\; 0$

Let

$$(Q^{opt}(q/r), \ell^{opt}(q/r)) = \underset{Q,\ \ell}{\operatorname{argmin}}\, J(Q, \ell; m, r, q)$$

and let $p(Q)$ be the number of eigenvalues of Q on the unit circle.

Theorem

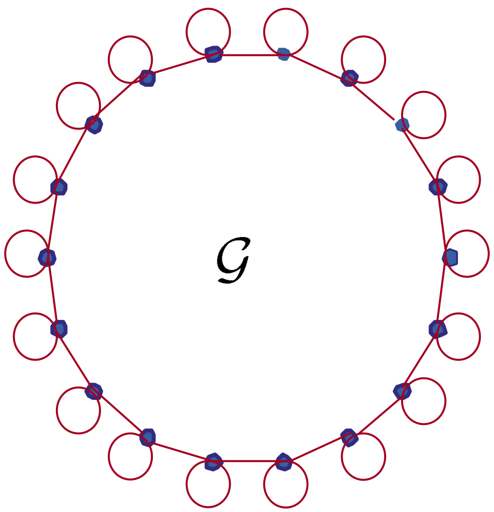
$$\lim_{q/r \to 0} p(Q^{opt}(q/r)) = \min_{Q} p(Q) =: p^{opt}.$$

Moreover

$$\ell^{opt}(q/r) = \sqrt{\frac{N}{p^{opt}}}\sqrt{\frac{q}{r}} + o\left(\sqrt{q/r}\right).$$

Simulation results: circulant graph

Consider the communication graph $\mathcal{G}$ with consensus matrix Q_k. Only two parameters to optimize: k and ℓ.

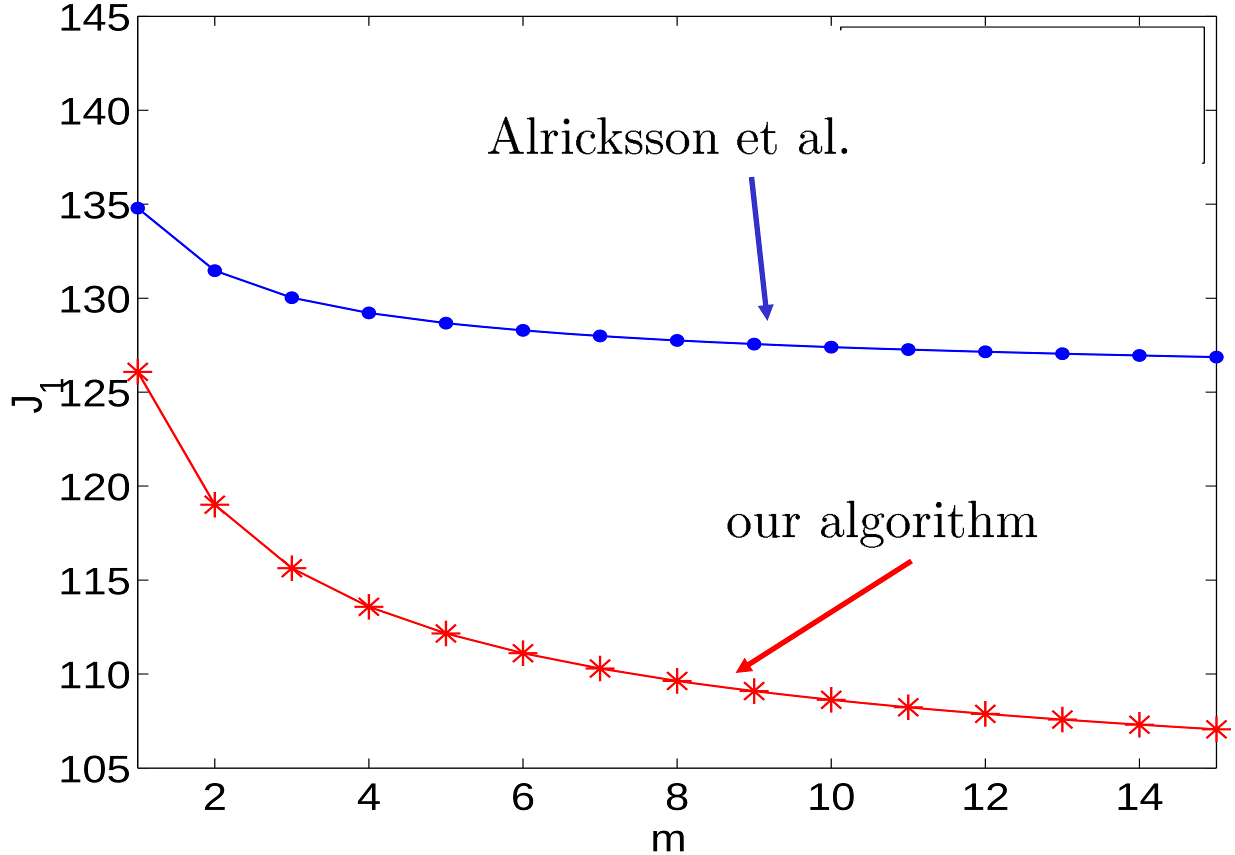


$$Q_k = \begin{bmatrix} 1-2k & k & 0 & \cdots & 0 & k \\ k & 1-2k & k & \cdots & 0 & 0 \\ 0 & k & 1-2k & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & k & 1-2k & k \\ k & 0 & 0 & \cdots & k & 1-2k \end{bmatrix}$$

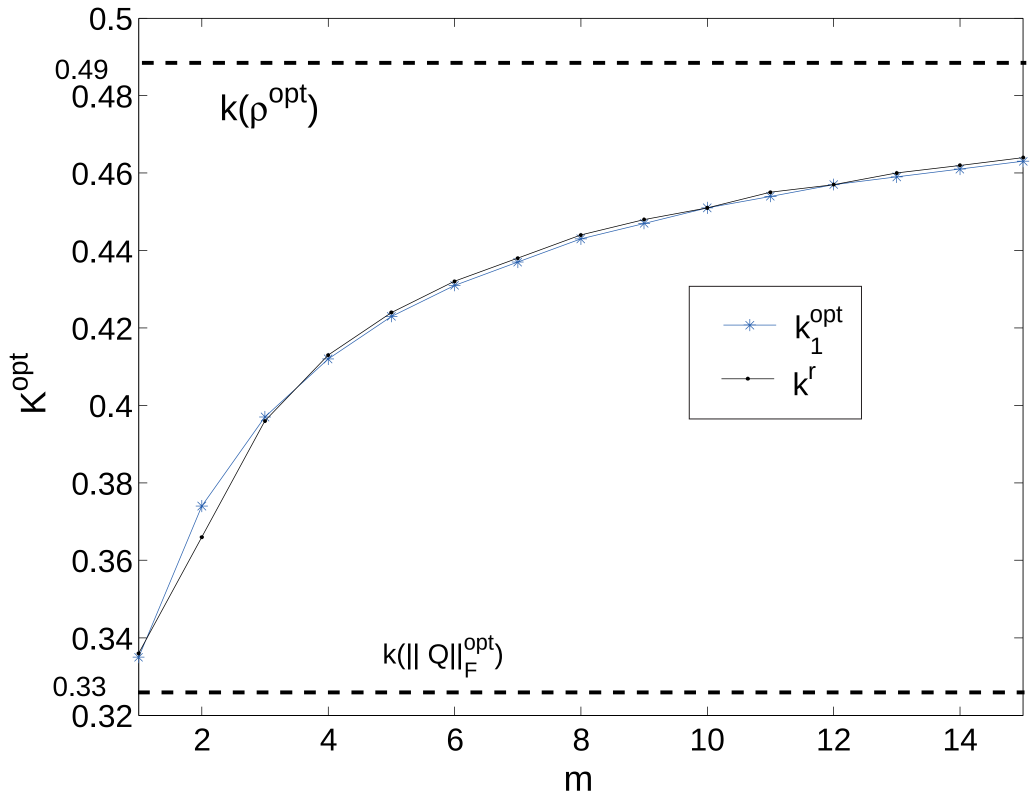
We assume that N=100, q=1 and r=1.

Simulation results: Circulant graph

J_1^r refers to the approach by Alriksson and Rantzer [06]: *same* cost, but $\ell_i(t)$ and $Q(t)$ computed *recursively* at each time step (more general setup: multivariable dynamics).



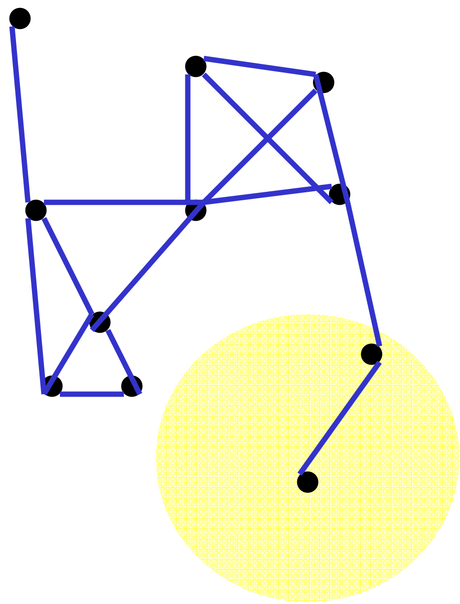
Simulation results: Circulant graph

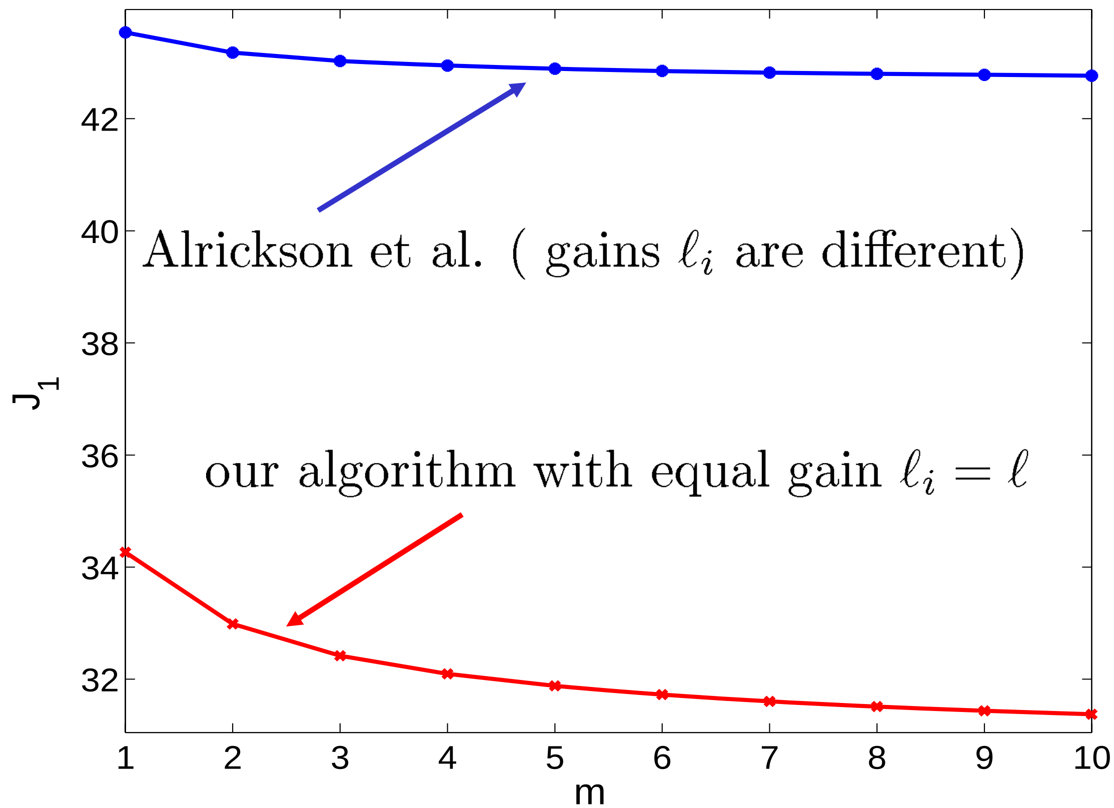


For slow comminication ($m \approx 1$) it is better to optimize $||Q||_F$ than $\rho(Q)$.

REMARK: $||Q^m||_F = 1 + c\rho^m(Q) + o(\rho^m(Q))$

Simulation results: Random geometric graph





Not possible to minimize $P(t)$ but only $trace(P(t))$ (in centralized Kalman they are equivalent).

Takeaways points

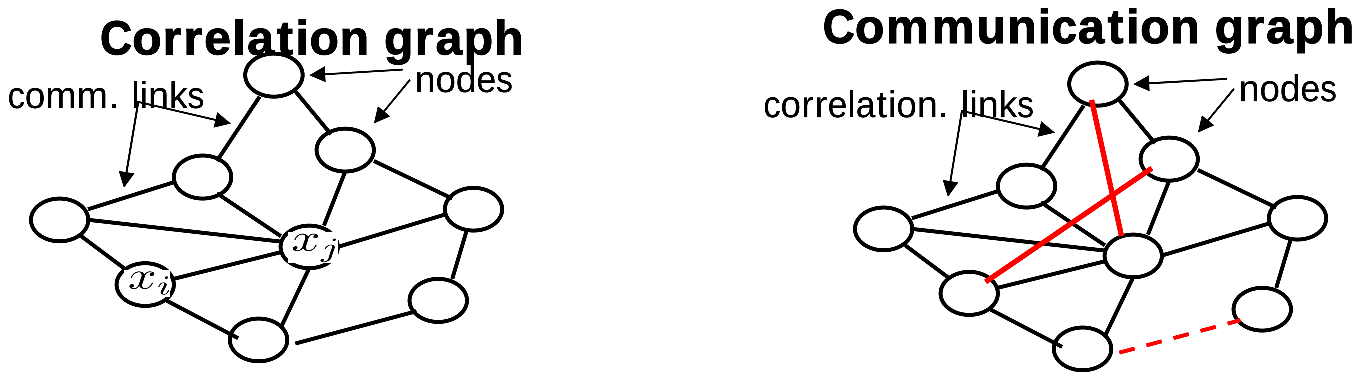
- Consensus algorithms fit naturally in distributed estimation problems
- Some analytical results for scalar dynamics under special regimes
- Optimizing second $\lambda_2(Q)$ is not necessarily optimal strategy

Outline

- Motivations and target applications
- Overview of consensus algorithms
- Application of consensus to WSN:
  - Sensor calibration
  - Least-square parameter identification
  - Time-synchronization
- Open problems
  - Identification
  - Estimation
  - Control

Identification: large scale structured systems

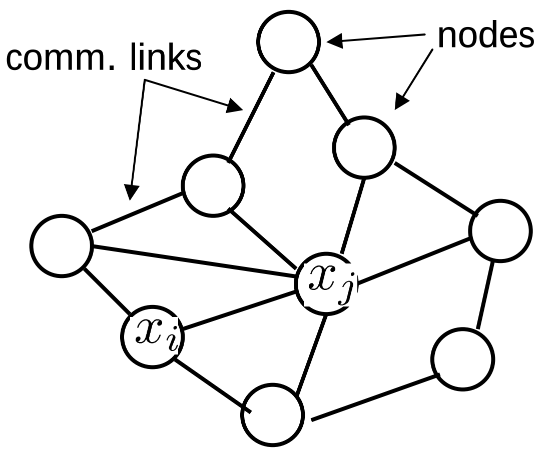
$$x \sim \mathcal{N}(0, \Sigma), \quad \Sigma^{-1} \text{ is sparse (graph model)}$$

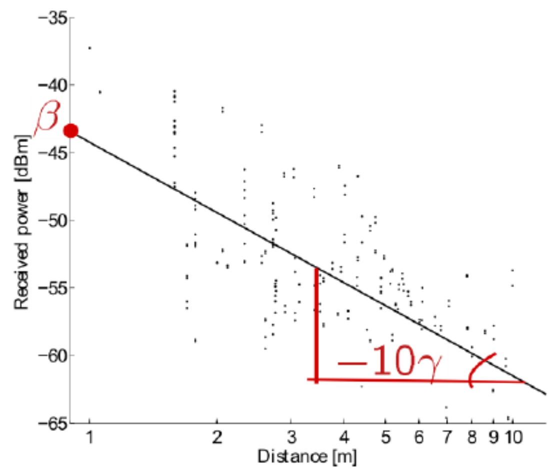


- Σ only partially known and noisy $\Rightarrow \Sigma^{-1}$ is full.
- communication graph $\neq$ correlation graph
- weak correlation, i.e. Σ^{-1} full w/ some small entries $\Rightarrow$ Graph identifiability
- what if dynamics also, i.e. $x_{t+1} = Ax_t + w_t$?
- if a node dies, i.e. remove row-column from Σ, how to compute Σ^{-1} ?
- how to do model reduction preserving graph structure ?
- is consensus relevant ?

Carlos Carvalho " *Structure and Sparsity in High-Dimensional Multivariate Analysis*", Ph.D. Theis, Duke Univ., 2007
A. P. Dempster, *Covariance Selection*, Biometrics, Vol. 28, No. 1, Special Multivariate Issue (Mar., 1972), pp. 157-175

Non-parametric estimation





Identification/Estimation of infinite dimensional space $f : R^n \rightarrow R$.

Centralized learning: $\hat{f}(\cdot) = \sum_{n=1}^{N} \alpha_n \Phi(x_n, \cdot)$

Totally decentralized learning: $\hat{f}_i(\cdot) = \sum_{n=1}^{N_i} \alpha_n^i \Phi(x_n, \cdot), \quad \mathbf{N_i << N}$

kernel

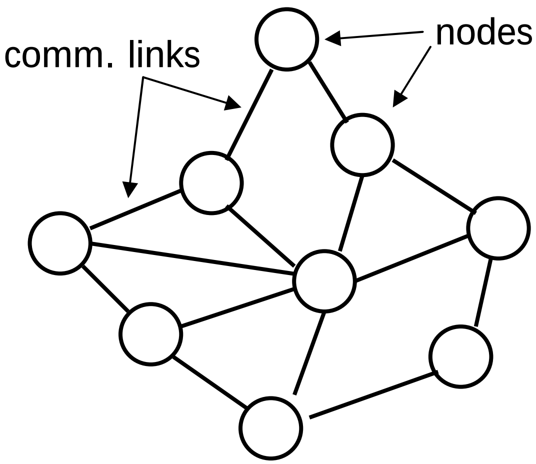
What to exchange ?

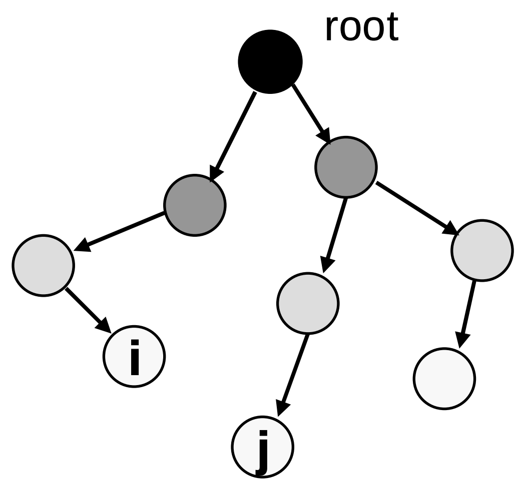
- all $(x_i, f(x_i))$ of neighbors ?
- most informative $(x_i, f(x_i))$ of neighbors ?
- smoothed observation of neighbors $(x_i, \hat{f}_i(x_i))$
- virtual observations $\hat{f}(\hat{x}_i, \hat{f}_i(\hat{x}_i))$

Predd, J.B.; Kulkarni, S.B.; Poor, H.V. "*Distributed learning in wireless sensor networks*" Signal Proce Magazine,2006

Soft Hierarchical Control

Time synchronization example:





P_{dist} symmetric:
slow convergence but robust

P_{hier} asymmetric:
fast convergence but fragile to node failure

$$\boxed{P_{soft} = \alpha P_{dist} + (1-\alpha) P_{hier}, \qquad \text{optimal } \alpha \text{ depends on failure rate}}$$