SYNCHRONIZATION in SENSOR NETWORKS

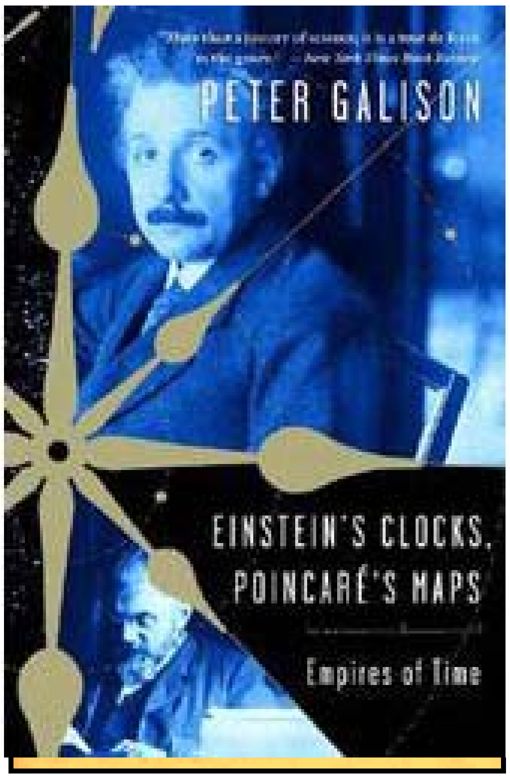
How does network topology affect behavior?

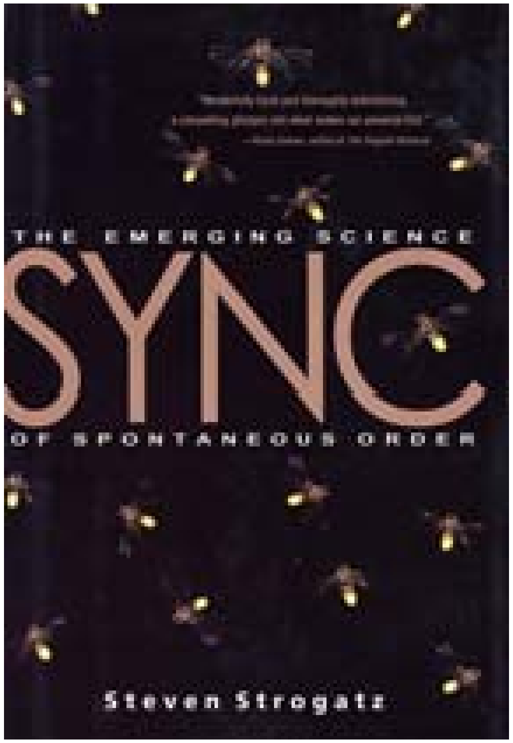
Ananthram Swami
U.S. Army Research Lab
a.swami@ieee.org

21 July 2008, SAM 2008

Acknowledgements:
Sergio Barbarossa, Prithwish Basu, Mounir Ghogho, Prabir Barooah, Joao Hespanha, Xiaoli Ma, Brian Sadler, Arnt Salberg, Anna Scaglione

Sync: An old problem!

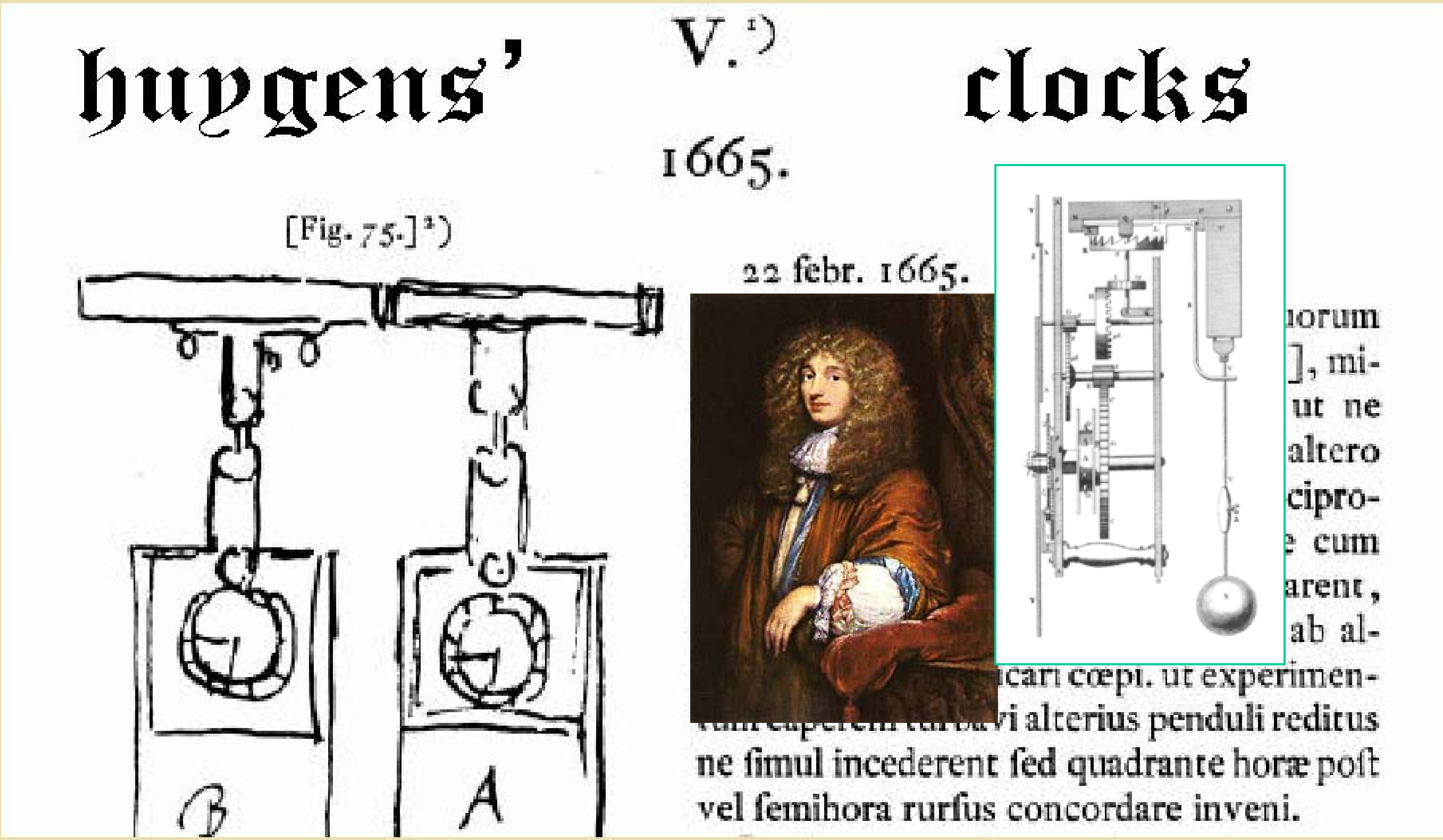




Poincare: 1909 Broadcast radio sync

1884 – World Time Conference

Sync: An old problem!



Sync: An older solved problem?

Kaempfer, 1680; Kuaramato, 1984; Mirollo-Strogtaz, 1990

- ***~10K male Southeast Asian fireflies congregate in trees and flash in near perfect unison ****
  - ***flash every 800 to 1600 ms***
  - ***Sync to within 30 ms***
  - ***Only local interaction: leaves, limited perception etc.***
  - ***flash every ~1s, without external stimulus.***
- ***Knowledge of distributed mechanism used to synchronize flashes can be used to develop better ways to synchronize on a network***

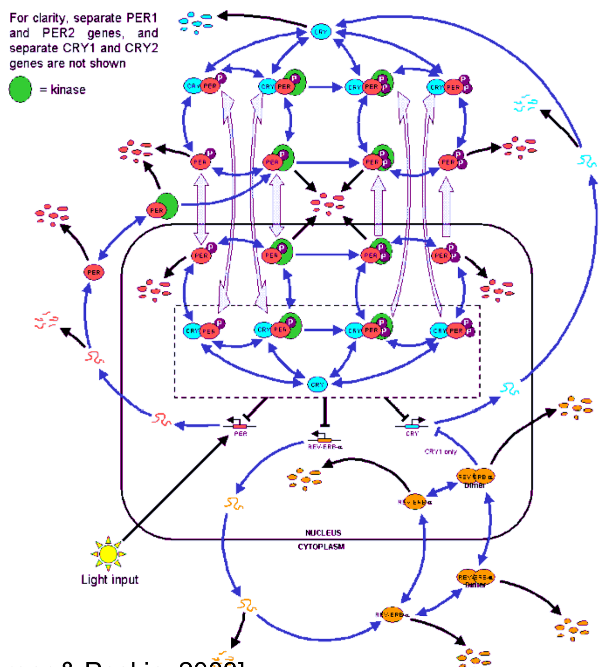
Photo from presentation by: Ibiso Wokoma, Ioannis Liabotis, Ogngen Prnjat, Lionel Sacks, Ian Marshall – University College of London

Source: Talk by Dr. John Parmentla, Director for Research and Laboratory Management, US Army, at USMA Network Science Workshop, on 22 Oct 2007.

Sync: An older solved problem?

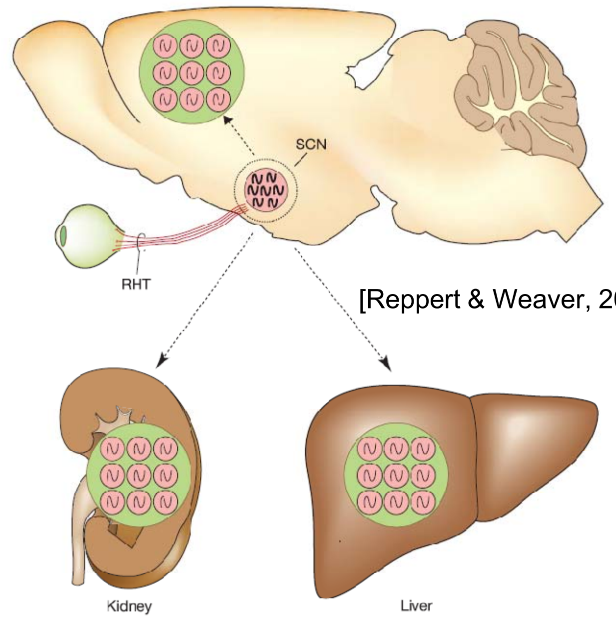
Source: Talk by Dr. John Parmentla, Director for Research and Laboratory Management, US Army, at USMA Network Science Workshop, on 22 Oct 2007.

Clock Neurons in the Suprachiasmatic Nucleus (SCN)

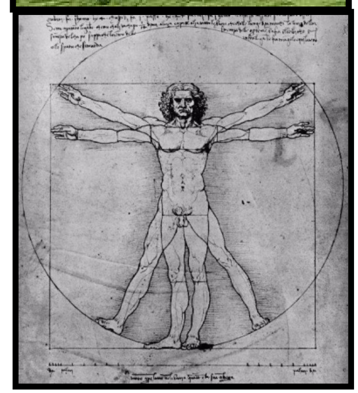


[Forger & Peskin, 2003]

- *Eyeball and brain slice*



[Reppert & Weaver, 2002]



Gene networks* ⟶ *Organs (~10,000 cells)* ⟶ *Organism

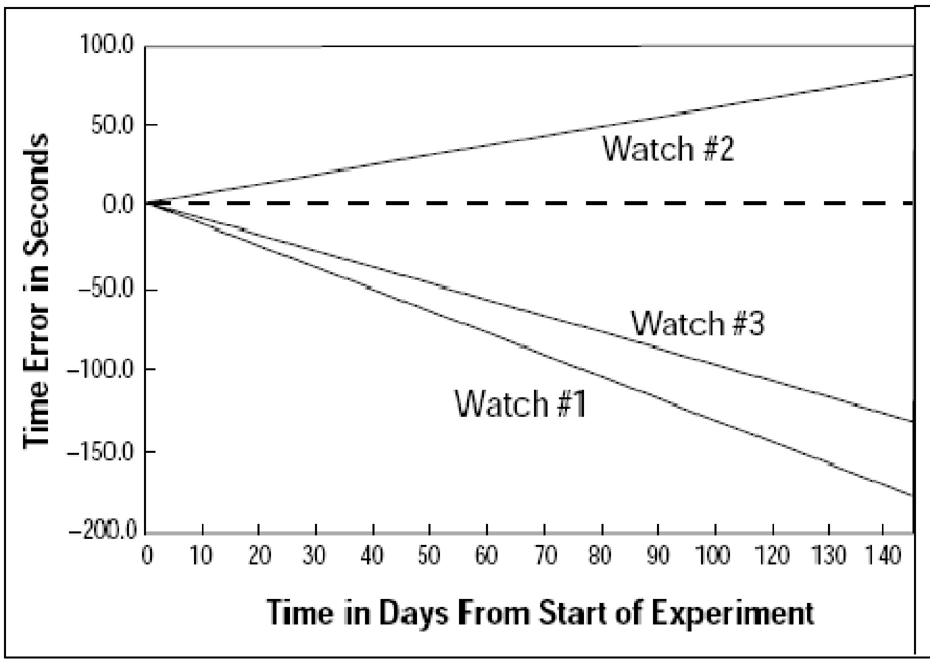
Individual neurons are sloppy timekeepers - but synchronized neurons are precise clocks

Source: Frank Doyle, ICB, University of CA at Santa Barbara

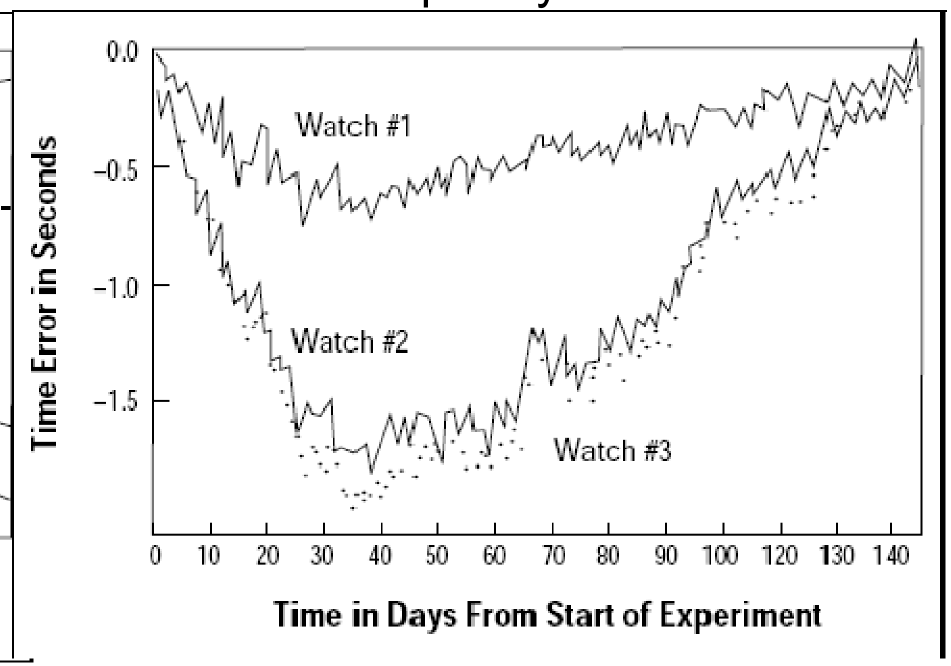
Example

❖ Experiment
  – 3 watches placed next to each other
  – Left for 140 days, time recorded each day

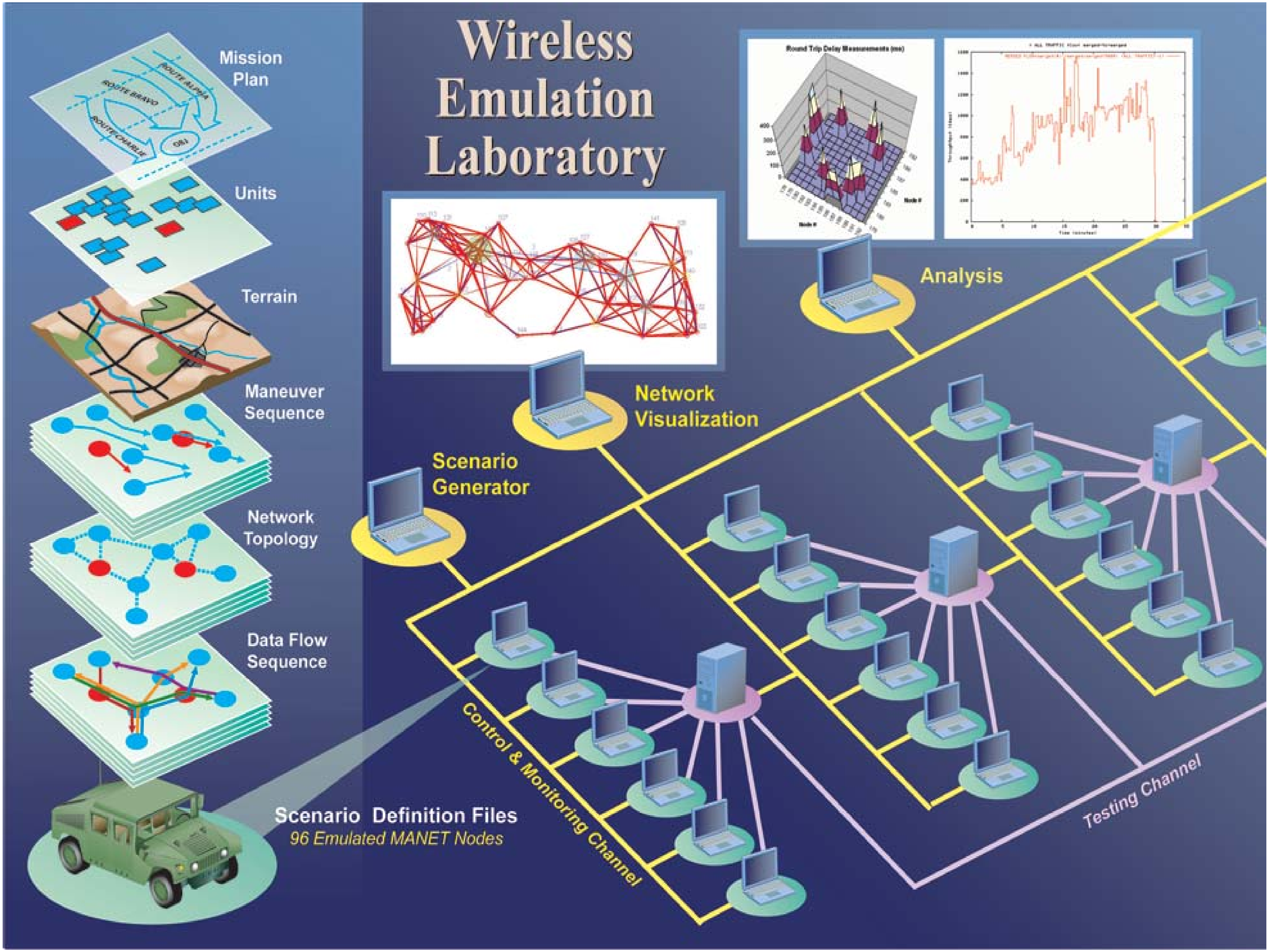
Uncorrected Time



Time With Frequency Correction



Source: From a CTA talk by Prof Xiaoli Ma, Georgia Tech;
taken from `The Science of Timekeeping', HP Application Note 1289


Wireless Emulation Laboratory
Mission Plan
ROUTE ALPHA
ROUTE BRAVO
ROUTE CHARLIE
OBJ
Units
Terrain
Maneuver Sequence
Network Topology
Data Flow Sequence
Scenario Definition Files
96 Emulated MANET Nodes
Round Trip Delay Measurements (ms)
Node #
Analysis
Network Visualization
Scenario Generator
Control & Monitoring Channel
Testing Channel

Clock Accuracy

	Accuracy (PPM)	Power	Lifetime with AAA battery 1250 mAh	Notes
Watch clock	200×10^{-6}	1 micro W	142 yrs	Temperature, aging
TCXO	6×10^{-6}	6 mW	208 hours	>1 PPM
MCXO	3×10^{-8}	75 mW	17 hours	large, aging drift
GPS	10^{-8} -- 10^{-11}	180 mW	7 hours	Outdoors, cost
DARPA CSAC	10^{-11}	30 mW 125 mW	42 hours 10 hours	*Target* *prototype*

Caveats:

- Battery lifetime depends upon discharge current
- Energy storage depends upon medium
- Drift to temperature, aging → must resync
- Resync time ≈ (Tolerable offset) / (relative accuracy)

Motivation

Time-Sync Crucial for

- Target tracking, ranging, localization
- Distributed MIMO, Collaborative signal processing
- Data fusion
- Feedback control
- Network probing and monitoring
- Cooperative communications
- Energy efficient MAC (e.g., with duty cycling)

Local information

- But need to estimate `global' state

Constraints

- Resource constraints (batteries, BW)
- Application constraints (timeliness, desired accuracy)

Robustness

- Link and node failures; asymmetric comms; variable delays

Exemplar for distributed inference

Metrics & Tradeoffs

- **Accuracy**: Worst case (or avg case) pair wise error between one-hop neighbors. Performance bounds?
- **Resource efficiency**: The number of broadcasts necessary to achieve sync and the rate and frequency of messages that need to be exchanged to maintain sync.
- **Convergence time**: The time taken for all nodes (or a high percentage of nodes) to sync to their neighbors.
- **Fault-tolerance**: Robustness to failure of critical nodes and/or links, clock jitter and drift, congestion, mobility.
- **Scalability with network size**: Does the sync-error increase with size? Does the convergence time increase with the diameter of the network? Node density? Clock parameters?
- **Complexity**
- **Impact of variable delays** (queue, processing …)

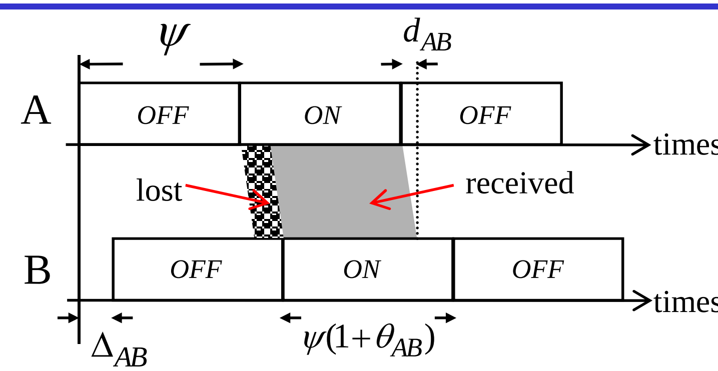
Some (recent) surveys on synchronization

Lindsey et al, Proc. IEEE, 1985
Bregni, IEEE Comm Mag, 1998

Anceaume, Puaut; INRIA, 1998
Sivrikaya, Yener, IEEE Network 2004
Johannessen, IEEE Contr. Sys Mag, 2004
Sundararaman et al, AHN, 2005
Sadler, Swami, MILCOM 2006
Faizulkhakov, PCS, 2007

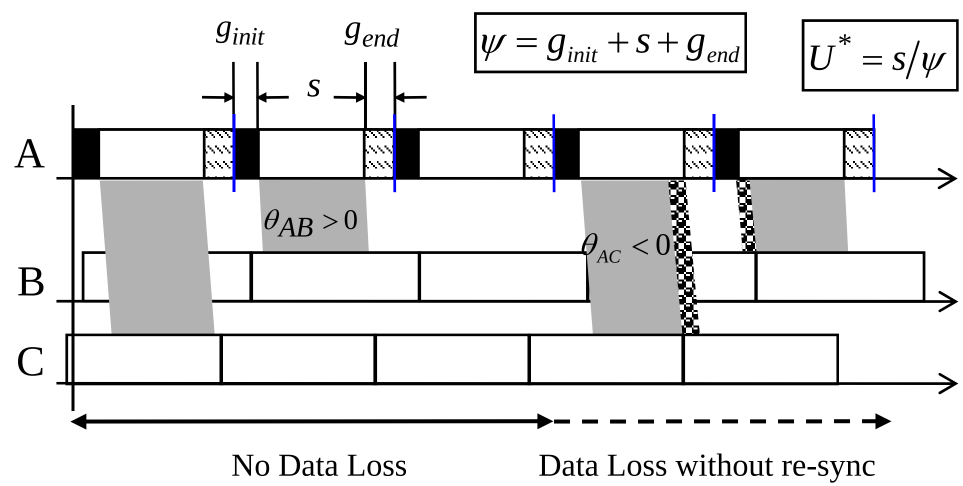
Slot Synchronization

- Motivation: Enabling energy conserving duty cycling MAC
  - Oscillator drifts and duty cycling
    - => misalignment of slot boundaries
    - => loss in throughput



Transmission from node A to B with loss.
B has a slower clock than A ($\theta_{AB} > 0$)

- Pair wise slot sync and re-sync
  - Requires guard times in each slot; and re-sync
  - Strong correlation between
    - worst case oscillator drift
    - re-sync period
    - slot utilization
    - energy consumption



Transmission from node A to B and A to C

(source: Dr. P. Basu BBN) (C&N CTA)

Slot-sync Schemes: Tradeoffs

Dimensions: Sync accuracy, Energy efficiency, Convergence Time, Fault tolerance, Scalability, Engineering Simplicity

Slot-sync Scheme	Advantages	Disadvantages
Tree based – BFS (driven by periodic Heartbeats)	Fast convergence after tree computation (can be reused or amortized); Low #local broadcasts for sync maintenance; Provably low worst-case sync-error	Fault-tolerance hard to achieve in duty-cycle mode (tree needs to be recomputed); If root fails, then re-election is hard; Tree maintenance could costs energy
Tree based – BFS (driven by Link State Updates)	Sync-trees with better properties hence lower energy for maintaining sync	Repairs may take a long time because of the dependence on LSUs (low frequency)
Root based – MinDelay (driven by periodic Heartbeats)	Faster convergence because the tree structure is only implicit after root election; Better fault-tolerance than other tree based schemes	Worst-case sync-error may be worse than other tree based schemes; Also energy consumption (#broadcasts) for maintenance may be high
Peer-to-peer based on aggregation/filtering (driven by periodic Heartbeats)	No root election or sync-tree computation step, hence highly fault-tolerant; Sync maintenance is very efficient; Friendly to network dynamics (join/leave/move); Potentially better scaling properties	Slower convergence than tree-based schemes; Potentially lower accuracy; Determining optimal aggregation function (median? mean? other?) may be hard
Hybrid (tree-based and peer-to-peer)	Best of both worlds: achieve fast convergence by tree-based protocol followed by fault-tolerant maintenance of sync in low-energy mode	Two phase protocol – hence may be harder to analyze and implement

(source: Dr. P. Basu BBN) (C&N CTA)

Time Sync Protocol Taxonomy

- **Broadcast Protocols**
  RBS *Elson, Girod, Estrin, 2002:* Beacon-aided
  TPSN, *Ganeriwal et al, 2003,* Broadcast with hierarchy

- **Distributed Synchronization Protocols**
  Diffusion-based: *Li, Rus, 2004*
  Spatial smoothing: *Solis et al, 2006*
  Bio-inspired *Hong, Scaglione, 2005*
  Average consensus: *Xiao, Boyd* 2003
  Jacobi iterations: *Barooah et al, 2007*
  Advection-diffusion *Barbarossa, 2008 (here!)*

Other taxonomies:

- Single-hop vs. multi-hop
- Server initiated vs. client initiated vs. always-on

Clock Models

- t = "true" time, local node time is *T(t)*
- Clock drift $\rho(t) = dT(t) / dt - 1$
- Bounded drift, and clocks not running backwai

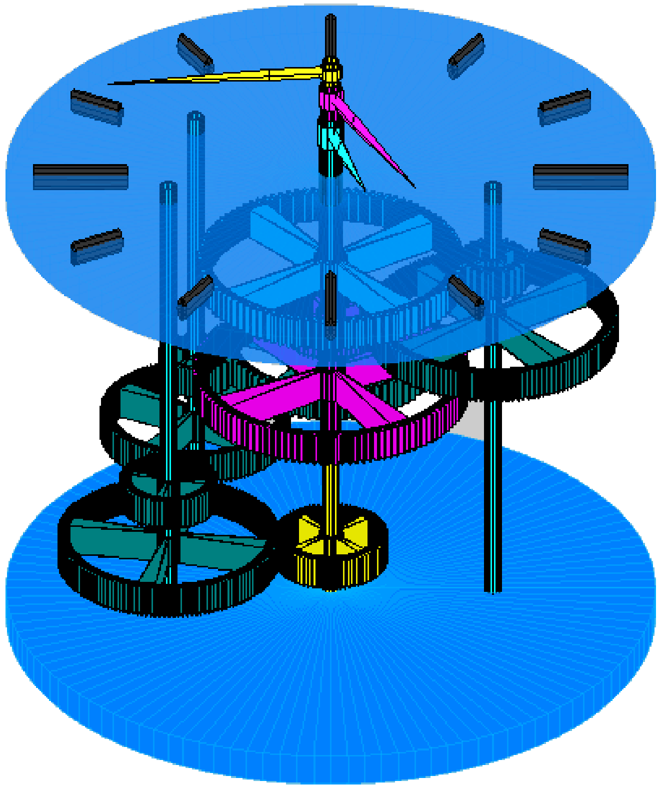
$$|\rho(t)| < \rho_o \qquad -1 < \rho(t)$$

- Taylor expansion at the i^{th} node

$$T_i(t) = \alpha_i + \beta_i t + \gamma_i t^2 + \ldots$$

offset *skew* *models time variations*

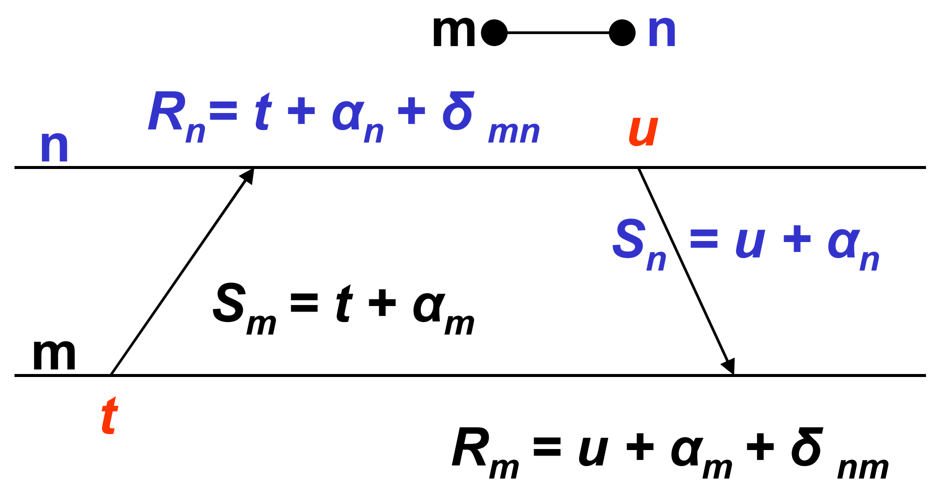
- Could model skew as AR process
- Time Sync problem: Estimate α, β, γ, and adjust clock



Pair-wise synchronization

Estimate relative time offsets

- Exchange time-stamped packets
- Mark transmit & receive times
- Extends to skew estimation



$$2\zeta_{mn} = R_n - S_m + S_n - R_m = 2\,(\alpha_n - \alpha_m) + (\delta_{mn} - \delta_{nm})$$

$$\zeta_{mn} = (\alpha_n - \alpha_m) + \varepsilon_{mn}$$

Time shifts from reference clock

Zero mean error

Same formulation in estimating heading, position, skew

Use multiple measurements.
Linear least-squares problem:

- ➢ Pair-wise sync easy
- ✓ Can easily compute variance CRB, robust estimators confidence intervals etc.
- ✓ Test for drift $\beta \neq 0$?

Pair-wise estimation of skew and offset

➢ Node u transmits a pair of packets spaced by ρ

Measurements

$$
\begin{aligned}
R_2(k) &= \alpha S_1(k) + \beta + \delta\alpha_2 + \epsilon_{12}(k) \\
S_2(k) &= -\alpha R_1(k) + \beta - \delta\alpha_2 + \epsilon_{21}(k) \\
\bar{R}_2(k) &= \alpha(\rho + S_1(k)) + \beta + \delta\alpha_2 + \bar{\epsilon}_{12}(k)
\end{aligned}
$$

Estimates

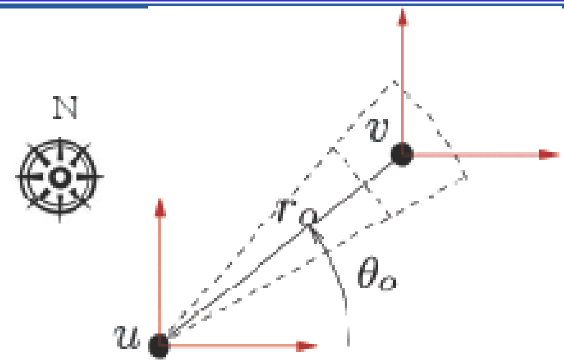
$$
\begin{aligned}
\hat{\alpha} &= \frac{\bar{R}_2 - R_2}{\rho} \\
\hat{\beta} &= -\frac{\hat{\alpha}}{2}(R_1 + S_1) + \frac{1}{2}(R_2 + S_2) \\
\widehat{\alpha_2\delta} &= \frac{\hat{\alpha}}{2}(R_1 - S_1) + \frac{1}{2}(R_2 - S_2)
\end{aligned}
$$

Covariance

$$
\frac{\sigma_w^2}{2\rho^2}\left[\begin{array}{c} 4 \\ (S_1 + R_1 + \rho)^2 - \rho(R_1 + S_1) \\ (S_1 - R_1 + \rho)^2 + \rho(R_1 - S_1) \end{array}\right]
$$

- Extends to multiple measurements.
- Usual techniques to deal with NG noise

Example: Localization

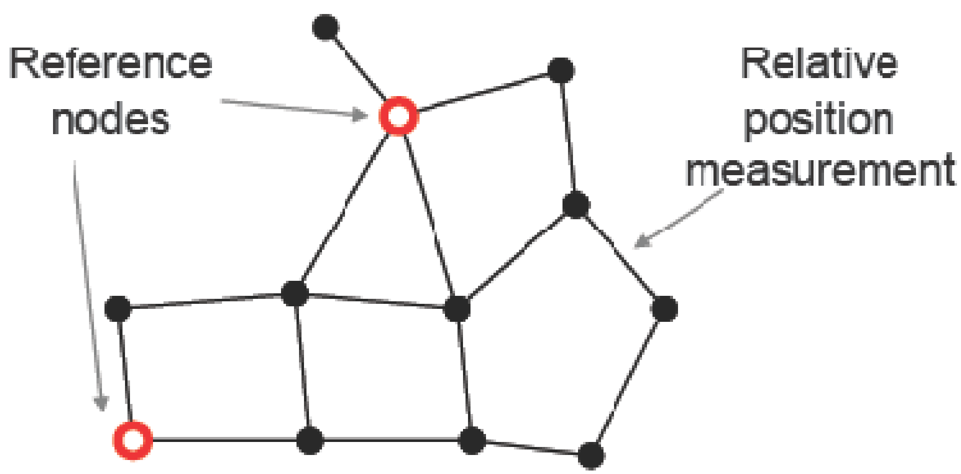


$$\zeta_{uv} = (\mathbf{x}_u - \mathbf{x}_v) + \epsilon_{uv}$$

Noisy measurement — Node position vectors — Zero mean random vector

- Every sensor can measure the relative positions of its neighboring sensors (dx and dy from r and θ).
- one or more node location(s) known.
- Local coordinates frames are aligned (nodes have compasses).

Problem: determine node positions (w.r.t. reference)



[source Dr. Prabir Barooah, U. Florida (ICB)]

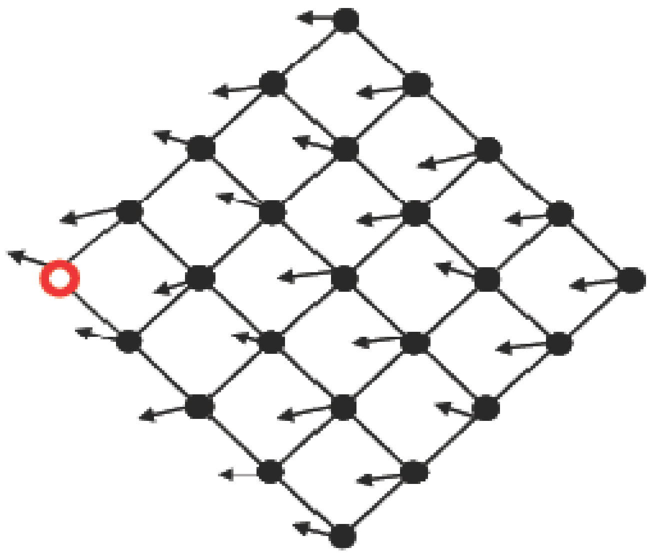
Traditional localization

- Only range measurements
- Only angle measurements

Recent interest in using both types of measurements

- "Ad-hoc localization using ranging and sectoring" Chintalpaudi, Dhariwal, Govindan and Sukhatme Infocom'04

Example: Motion coordination



$$\zeta_{uv} = (\theta_u - \theta_v) + \epsilon_{uv}$$

Headings with respect to reference heading

Zero mean error

- Every agent measures the difference in the heading between itself and nearby agents.
- One (or more) agent's heading is known (reference).

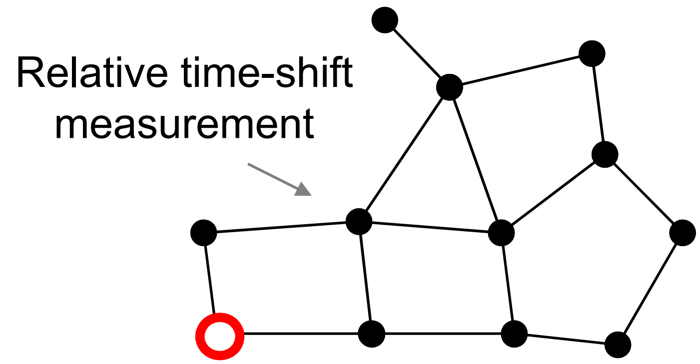
Problem: determine headings (w.r.t. reference headings)

[source Dr. Prabir Barooah, U. Florida (ICB)]

Network time synchronization

$$\zeta_{mn} = (\alpha_n - \alpha_m) + \varepsilon_{mn}$$

Same formulation in
- skew-offset estimation
- heading estimation
- position estimation



How to go from pair-wise to network-wide sync?

Incorporate info from `r' reference nodes?

Additive ambiguity : r > 0

Graph Representation

Relative Measurements

$$\zeta_{m,n} = \alpha_m - \alpha_n + \varepsilon_{mn}$$

Measurements **Offsets** **Errors**

Edge Vars **Node vars**

Measurements Graph

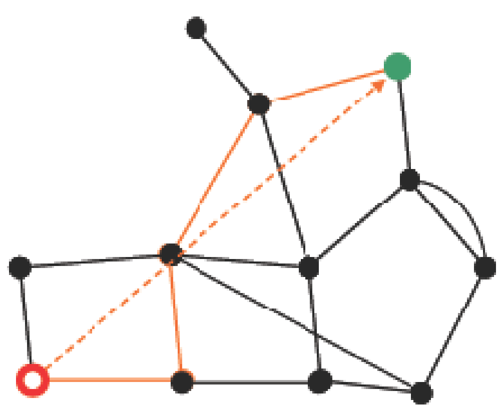
Communications Graph

G = (V,E)

$\zeta = \mathbf{A'}\,\alpha + \boldsymbol{\varepsilon}$

$\zeta = [\mathbf{A}_u' \, \mathbf{A}_r'] \, [\alpha_u \; \alpha_r] + \boldsymbol{\varepsilon}$

$\Rightarrow \mathbf{L}\,\boldsymbol{\alpha}_u = \mathbf{b}$



Estimators:

$L = A_u P^{-1} A_u'$ is invertible iff every weakly connected component in G has a reference node

Optimal estimate computed by FC with error-free links from all nodes

Performance

Covariance of optimal estimator depends on

- distance from reference node (`# hops')
- structure of the network

$$\Sigma_\alpha = L^{-1},\ \ L = A_u P_\varepsilon^{-1} A_u^T$$

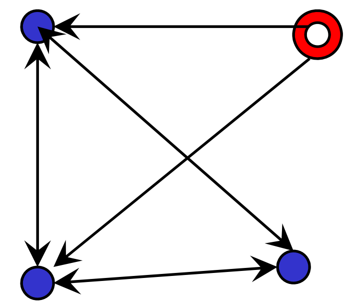
Distributed algorithms

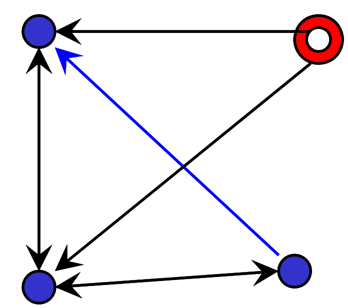
- Converges? To optimal ?
- Convergence rate
- Robustness to link failures ?
- Asymmetric communications ?
- With dynamics in topology ?

Distributed Algorithms: Assumptions

Assumptions

- Every weakly connected component of G = (V,E) has at least one reference node.
- Communication graph $G^c = (V, E^c)$: (u,v) in E $\Rightarrow$ (u,v) and/or (v,u) in E^c
- No edge in E^c is directed towards a reference node
- Measurement errors uncorrelated, known variances
- At every t, each node may fail with prob q, and every link with prob q.





Jacobi iteration for everv node $u \in \mathrm{V} \setminus \mathrm{V}_r$

$$\left(\sum_{v\in\mathcal{N}_u}\frac{1}{\sigma_{uv}^2}\right)\hat{x}_u^{(k+1)} = \sum_{v\in\mathcal{N}_u}\frac{1}{\sigma_{uv}^2}\left[\hat{x}_v^{(k)} + a_{uv}\zeta_{uv}\right]$$

Convergence?

Convergence Results

$$\left(\sum_{v\in\mathcal{N}_u}\frac{1}{\sigma_{uv}^2}\right)\hat{x}_u^{(k+1)} = \sum_{v\in\mathcal{N}_u}\frac{1}{\sigma_{uv}^2}\left[\hat{x}_v^{(k)} + a_{uv}\zeta_{uv}\right]$$

Weighted in-degree

$$D_{uu} = \sum_{v\in\mathcal{N}_u}\frac{1}{\sigma_{uv}^2}$$

Weighted adjacency

$$C_{uv} = \frac{1}{\sigma_{uv}^2}, \quad \text{if } (v,u)\in E^c$$

Submatrices of C & D

$$M,\ N \in \mathcal{R}^{n_u\times n_u}$$

Iteration:

$$M\mathrm{x}^{(k+1)} = N\mathrm{x}^{(k)} + \mathrm{b}_u$$

Fixed point
exists & is unique if
L_c is invertible

$$L_c = M - N = A_u^c P^{-1} A_u^T$$

Convergence Results - 2

Synchronous update, no node/link failures:

Matrix L_c is invertible iff there is a directed path in G_c from at least one reference node to every non-ref node.

Asynchronous case, with iid failures:

If $G_c(t)$ satisfies AS1; there is a directed path in G_{init} from at least one ref. node to every non-ref node; no communication edge in G_{init} fails permanently; no edge outside G_{init} remains active infinitely often. Then the algorithm converges a.s.

Proof follows Frommer & Syzld, On asynchronous iterations, Journal of computation & applied math, 2000; also Tsitsiklis and Bertsekas

Barooah, Hespanha, Swami, CDC 2007

Convergence Results – 3

There is a penalty in the asymmetric case: asymptotic covariances:

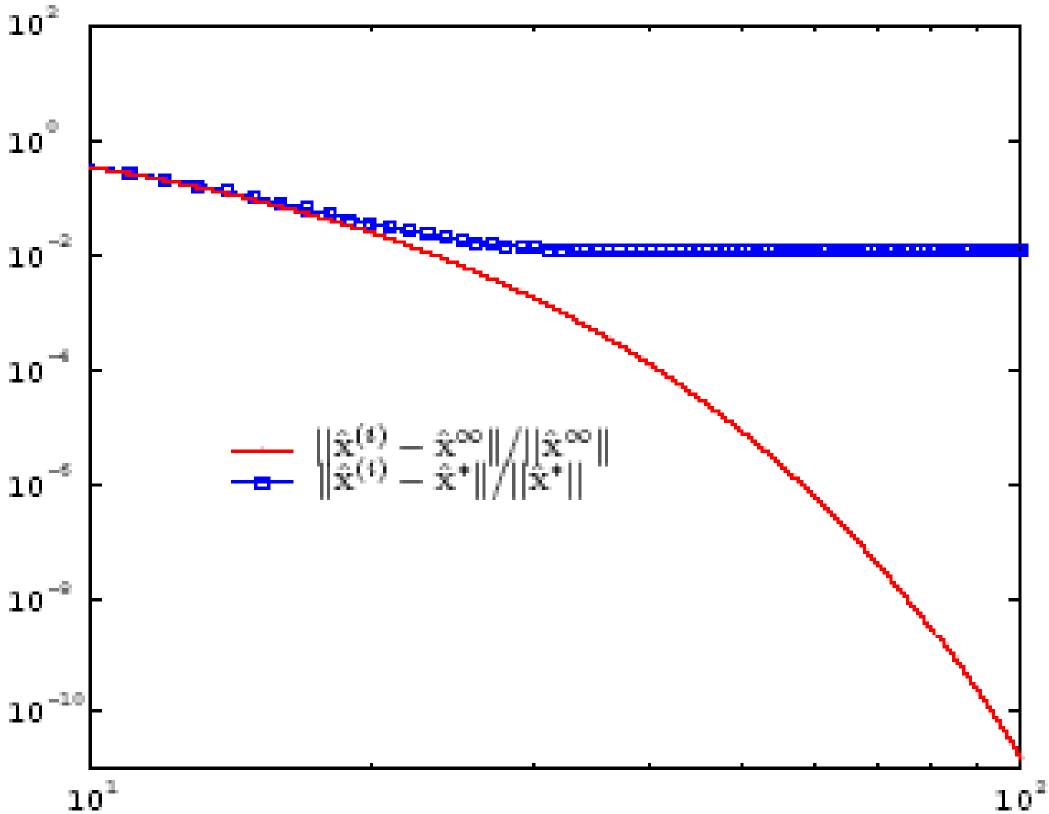
$$\Sigma_s = (A_u P_\varepsilon^{-1} A_u^T)^{-1}$$

$$\Sigma_a = (A_u^a P_\varepsilon^{-1} A_u^T)^{-1} (A_u^a P_\varepsilon^{-1} A_u^{aT}) (A_u^a P_\varepsilon^{-1} A_u^T)^{-T}$$

where A_u^a is obtained from A_u by setting appropriate elements to zero.

(u,e) = 0 if u is a reference node or the comm link e is not directed to u.

Simulation Results



Link failures p=0.2

No node failures q=0

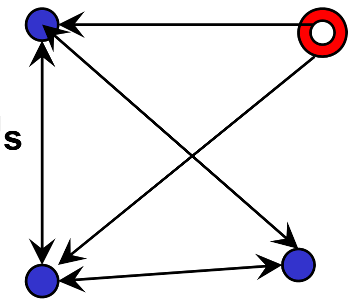
Estimate is unbiased

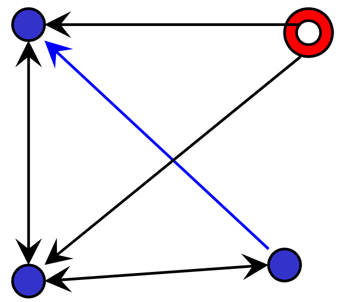
Converges rapidly

But with large variance compared with centralized BLUE

One link can be worth a lot!

G_s





G_a

1				-1
-1	1		-1	
	-1	1		1
		-1	1	

A

1				-1
-1	1		-1	
	-1	1		1
		-1		

A_a

0.625	0.375	0.5
0.375	**0.625**	0.5
0.5	0.5	**1**

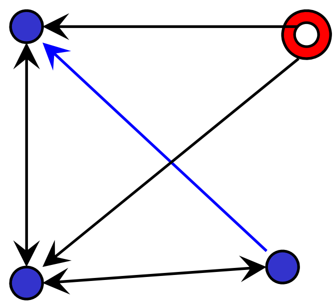
Σ_s

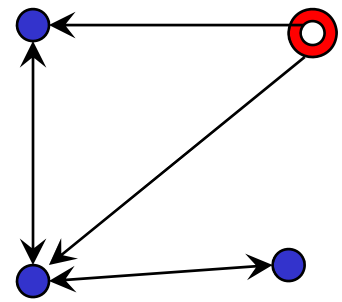
1.	0.75	1.25
0.75	**1**	1.25
1.25	1.25	**2.5**

Σ_a

Do more measurements help?

G1





G2

Var (G1) = [1, 1, 2.5]

Var (G2) = [2/3, 2/3, 5.3]

With dynamic topology changes

Iterations $x(t+1) = [I + D_t]^{-1} \quad [A_t + I] \, x(t) = F_t \, x(t)$

AS1 $F_t \, (i,i) >= a > 0$, $F_t \, (i,j) = 0$ or in $[a,1]$; unity row sums

AS2 Graph $G(t) = (N, E(t))$

Graph $(N, \text{Ú}_{t \leq s} E(s))$ is strongly connected for all $t >= 0$

AS3 Bounded inter-communication interval or symmetry

Then, the iteration converges to a common value

AS4. Bounded delays

AS5 With positive probability, some updates do not occur

Then, asymptotic consensus is achieved

Bondel et al (CDC 2005) / Tsitsiklis 1984:

Recursive Jacobi Algorithm

But Nodes can obtain `new measurements' during the iterations.

- Avg. relative offset with node v $\qquad \xi_{u,v}(i) = \frac{1}{i}\sum_{k=1}^{i} \zeta_{u,v}(i)$
- Update estimate wrt node v $\qquad \hat{x}_u^{(i)}(v) \;:=\; \hat{x}_v^{(i)} + \xi_{u,v}(i) \qquad \forall v \in \mathcal{N}(u)$
- Overall updated estimate $\qquad \hat{x}_u^{(i)} = (1-\beta_i)\hat{x}_u^{(i-1)} + \beta_i \frac{1}{d_u} \sum_{v \in \mathcal{N}_u} w_{u,v} \hat{x}_u^{(i-1)}(v)$
- In matrix form, for all nodes $\qquad \hat{\mathrm{x}}(i) = J\hat{\mathrm{x}}(i-1) + B\bar{\xi}(i-1)$

where

$$J = I - \beta M^{-1} L_b, \qquad B(i) = \beta M^{-1} A_b W,$$

$$\bar{\xi}(i) \;:=\; \xi^{(i)} - A_r^T \mathrm{x}_r, \qquad \xi^{(i)} = [\xi_1^{(i)}, \ldots, \xi_m^{(i)}]^T.$$

Barooah & Swami: MILCOM 2008 (submitted)

RJA – Convergence Results

$$\hat{x}_u^{(i)} = \alpha \hat{x}_u^{(i-1)} + \beta \frac{1}{d_u} \sum_{v \in \mathcal{N}_u} w_{u,v} \hat{x}_u^{(i-1)}(v),$$

1) If initial estimates $\hat{x}_u^{(0)}, u \in \mathcal{V}$ for the iterative update algorithm (8) are unbiased, the estimates $\hat{\mathrm{x}}^{(i)}, u \in \mathcal{V}$ are unbiased at every iteration i.
2) If the measurement noise $\{\epsilon^{(i)}, i = 1, \dots\}$ is a wide sense stationary white noise sequence with $R(i) = R_0, i = 1, \dots,$ then $P(i) \to P_\infty$ as $i \to \infty$, where the steady state covariance matrix P_∞ is the unique positive definite solution of the Lyapunov equation:

$$J P_\infty J^T - P_\infty + B R_0 B^T = 0. \qquad (12)$$

3) The spectra gap, assuming weak connectivity, is:

$$g(J) = 1 - \rho(J) = \beta \lambda_{\min} \left(M^{-1/2} L_b M^{-1/2} \right)$$

$$L_b := \bar{A}_b W A_b^T$$

M = diag (L_b)

$$J = \alpha I + \beta M^{-1} N,$$
$$B = \beta M^{-1} A_b W,$$

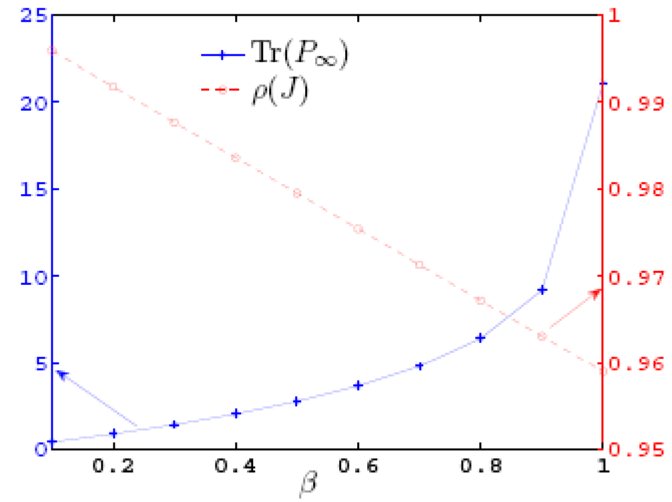


Fig. 1. The tradeoff in choosing larger β - it decreases the spectral radius (faster convergence rate) but increases the trace of the steady state covariance matrix P_∞ (poorer estimation accuracy).

Barooah & Swami : Milcom 2008 (submitted)

RJA – Simulation Example

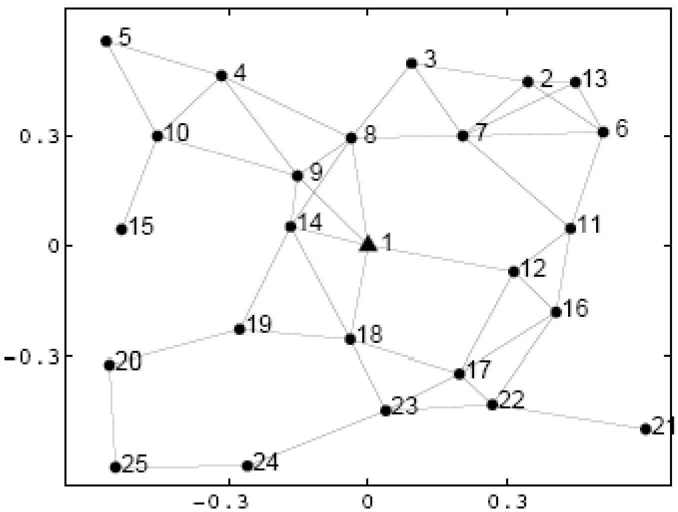


Fig. 1. The measurement graph for the 25 node simulation. All communi cations are bidirectional.

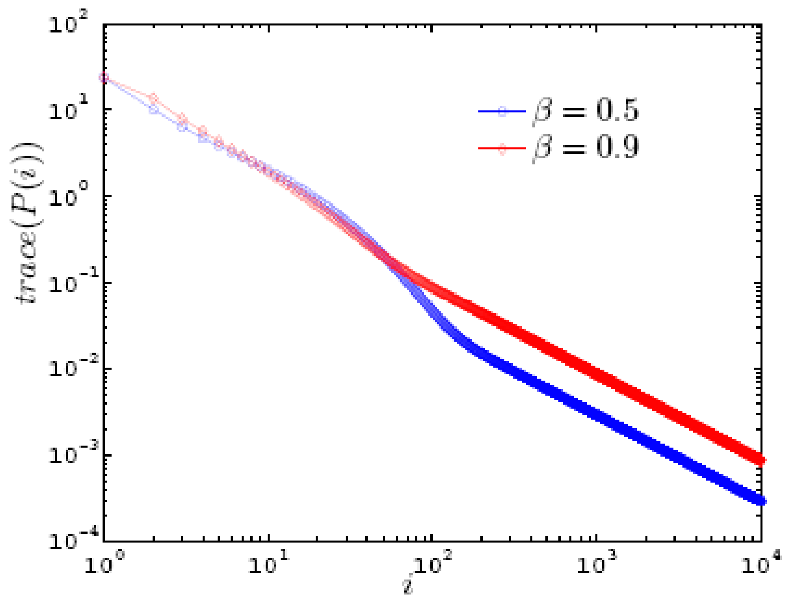


Fig. 2. The trace of the estimation error covariance matrix $P(i)$ as a function of the iteration counter i for the network shown in Figure 1. The covariance $P(i)$ is computed from the recursive relationship (12), with initial condition chosen as the identity matrix. Time evolution of the variances for two values of β are shown.

RJA – Simulation Example (2)

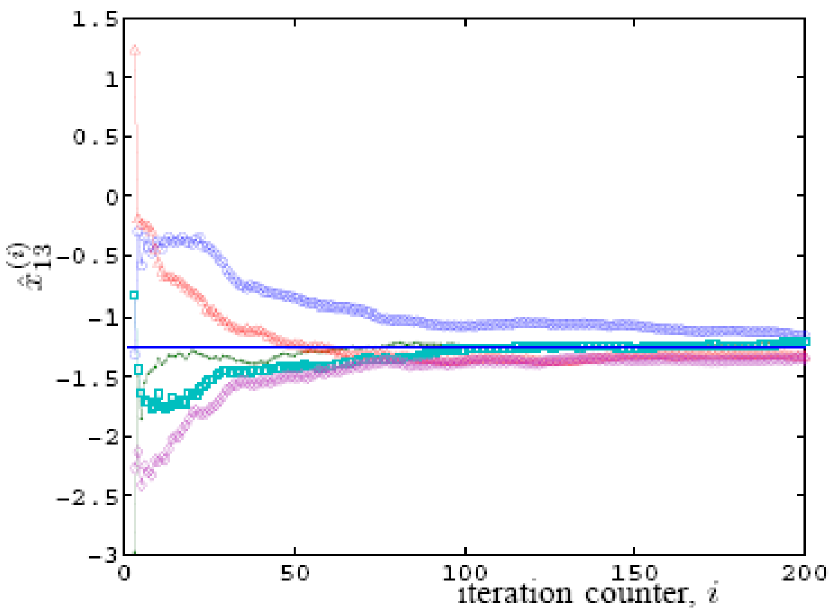


Fig. 3. Five sample runs of the estimates at node 13 (upper right hand corner node in Figure 1) as a function of time, with $\beta = 0.9$. The solid line shows the true value of node 13's time offset with respect to the reference.

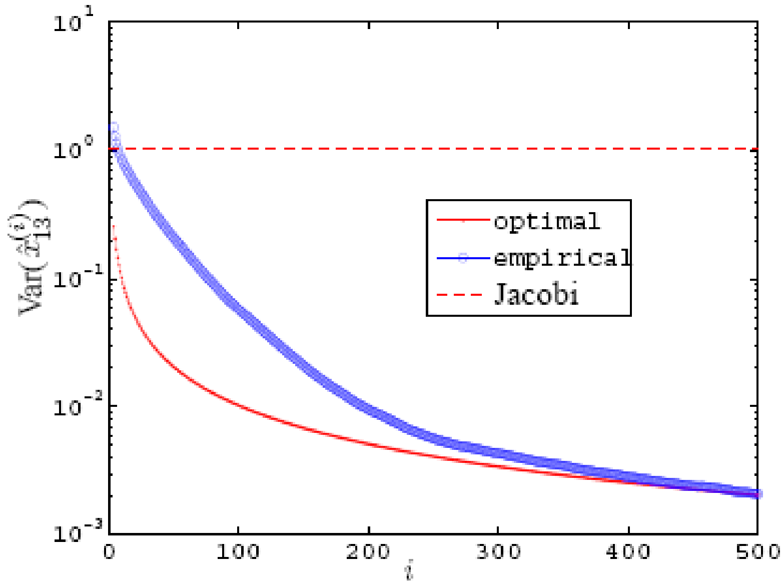


Fig. 4. The temporal evolution of the variance of the estimation at node #13 (shown in Figure 1). The legend "empirical" refers to the empirically estimated error variance of the node's estimates obtained by the proposed algorithm, with $\beta = 0.3$. The estimates were averaged over 100 sample runs. "Jacobi" refers to the steady-state variance the estimates produced by the Jacobi algorithm in [8] and its variants in [5], [10]

RJA – Simulation Example (3)

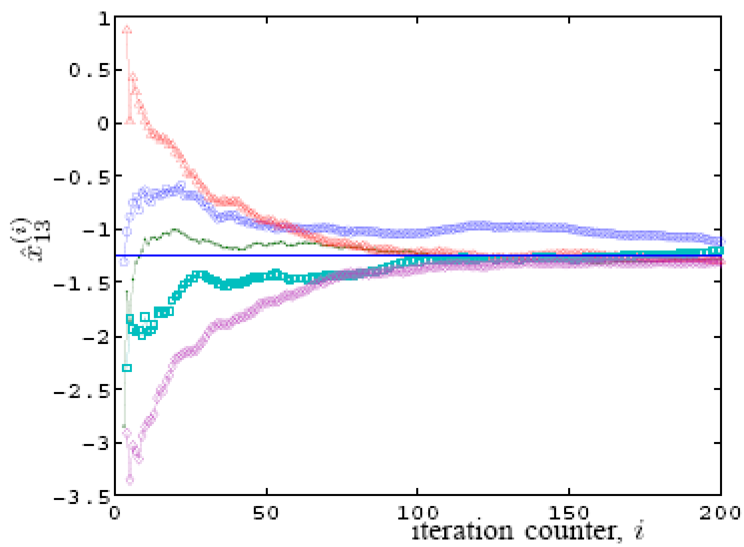


Fig. 5. Five sample runs of the estimates at node 13 (upper right hand corner node in Figure 1) as a function of time in the presence of random communication faults (probability of failure is 0.3), with $\beta = 0.9$. The solid line shows the true value of node 13's time offset with respect to the reference.

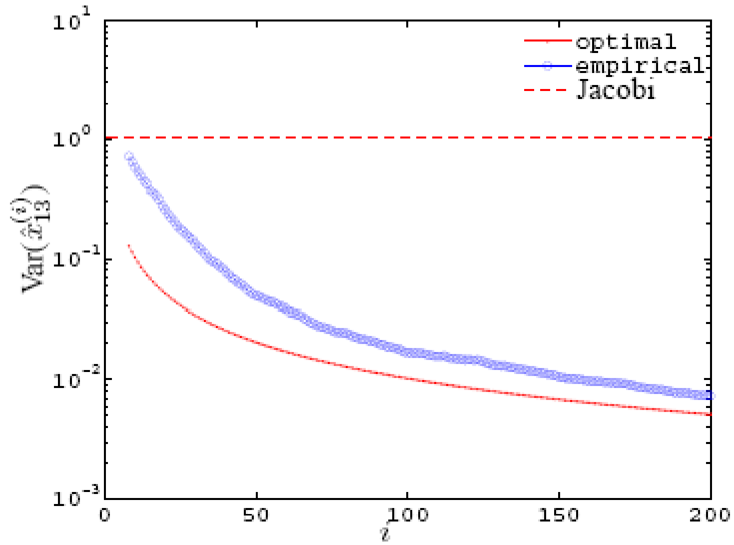


Fig. 6. Evolution of node 13's estimation error variance with random communication faults. At every iteration, every communication fails independently of all other edges, with a probability of 0.3. The legend "empirical" refers to the empirically estimated (from 100 sample runs) error variance of the node's estimates produced by the proposed algorithm. "Jacobi" refers to the variance the estimates produced by the Jacobi algorithm in [8] and its variants in [5], [10] achieve upon convergence without communication faults.

Convergence for grid graphs

Theorem 2: When all the edge weights are chosen as 1 (i.e., $w_{u,v} = 1$ for every $(u, v) \in \mathcal{E}$) and there is a single reference node $o \in \mathcal{V}$, the following statements hold.

1) In a 1-D grid of n_{total} nodes, we have

$$g(J) \geq \frac{\beta}{(n-1)(n-3)}$$

2) In a $N_1 \times N_2$ 2-D grid with a bounded aspect ratio (i.e., one in which there exist positive constants $\underline{c}, \overline{c}$ independent of N_1 and N_2 such that $\underline{c} \leq \frac{N_1}{N_2} \leq \overline{c}$),

$$g(J) \geq \frac{\beta}{n_{total}(\log n_{total} + \overline{c} + \underline{c} + 6)}$$

where $n_{total} = N_1 \times N_2$.

3) In a $N_1 \times N_2 \times N_3$ 3-D grid, where there exists positive scalars $\underline{c}, \overline{c}, \underline{d}, \overline{d}$ such that $\underline{c}N_1 \leq N_2 \leq \overline{c}N_1$ and $\underline{d}\log N_3 \leq N_2 \leq \overline{d}N_3^2$, we have that

$$g(J) \geq \gamma \frac{\beta}{n_{total}}$$

where γ is a constant independent of the number of nodes $n_{total} = N_1 \times N_2 \times N_3$. $\square$

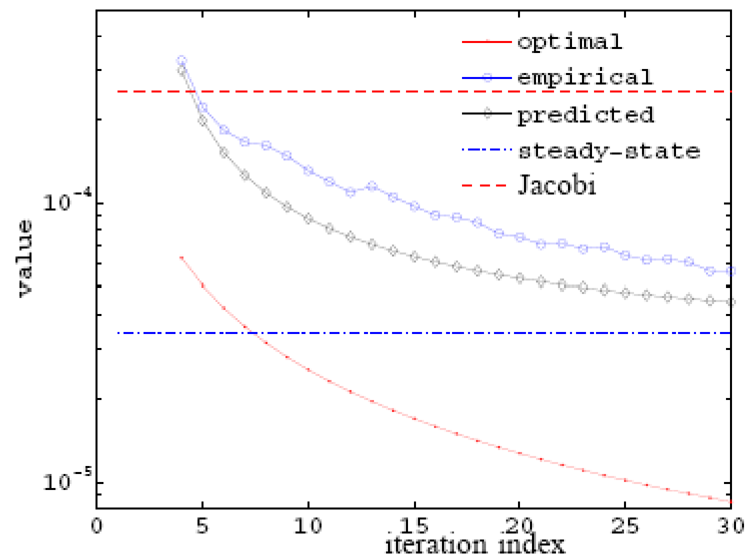


Fig. 3. The trend of several estimation error variances of the upper right hand corder node in the graph shown in Figure 2. The legend "empirical" refers to the empirically estimated (by Monte Carlo methods) error variance of the node's estimates produced by the proposed algorithm. "predicted" refers to the same variance, but that computed from the iteration (11). The legend "steady state" refers to the diagonal entry (corresponding to the node) of the steady state covariance matrix P_∞ described in Theorem 1. The trend of the variances corroborate the claims made in the Theorem. "Jacobi" refers to the variance the the estimates produced by the Jacobi algorithm in [6] and its variants in [4, 8] achieve upon convergence.

Summary

- Studied convergence of distributed (consensus) algorithms with asynchronous updates, iid failures, and asymmetric links.
- There is a performance penalty in the asymmetric case
- Variations on consensus algorithms to incorporate new measurements
- Adding measurements may be harmful:
  - ➢ Collaborative decision on which measurements should be added
  - ➢ Appropriate protocols
- ➢ When reference nodes disagree?

Anna Scaglione's Sync Video

The video clip was created by Prof. Anna Scaglione (Cornell / UC-Davis) and her group; see: http://www.youtube.com/watch?v=5F7Qhdf9ZJg

Questions?