César Asensio (cesar.asensio@uia.no)
University of Agder

April 20, 2018
Alberta, Canada

WISENET

Energy Efficient Consensus over Directed Graphs

Distributed Processing in IoT

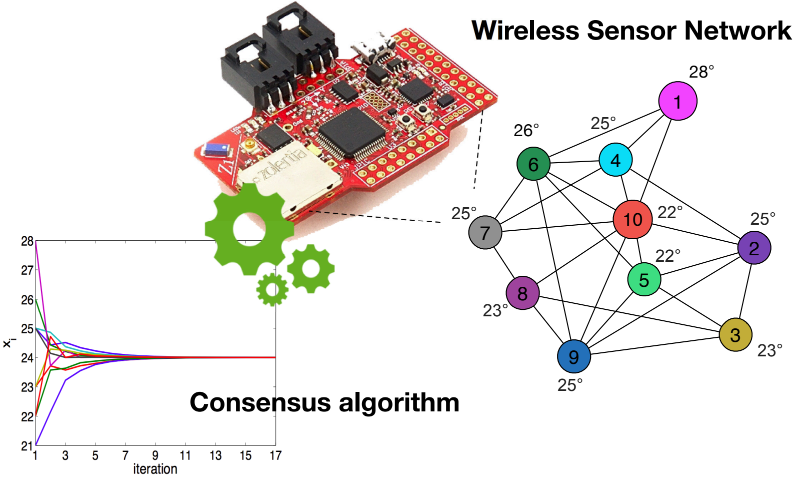
I Introduction and Motivation

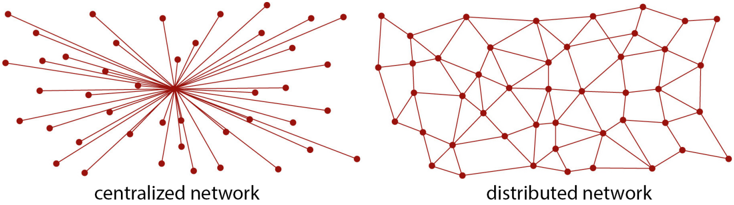
II Background on Consensus Algorithms

III Network Lifetime Maximization

IV Conclusions and Open Problems

Introduction and Motivation

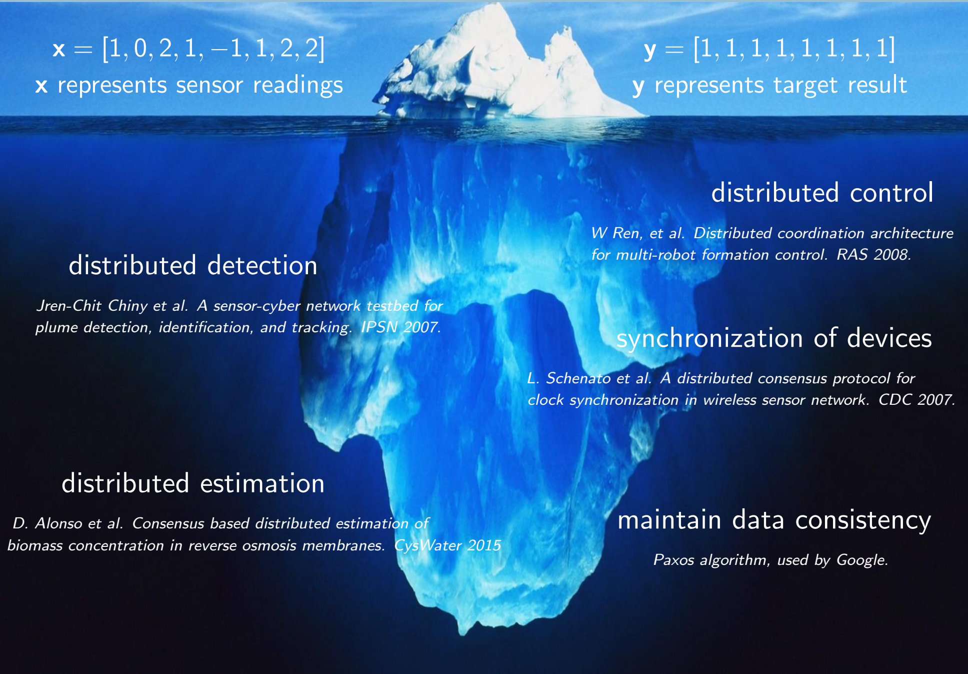

Wireless Sensor Network
Z1
zolertia
1
2
3
4
5
6
7
8
9
10
28°
25°
26°
25°
22°
25°
22°
23°
23°
25°
Consensus algorithm
x_i
28
27
26
25
24
23
22
21
1
3
5
7
9
11
13
15
17
iteration



- To avoid a single point of failure.
- To avoid congestion around central entities.
- It is cheaper.
- Allows incremental growth.
- Better scalability.

- Evolution of WSNs towards The Internet of Things (IoT) . . .
- resulting in apearance of heterogeneous sensor networks . . .
- where nodes present very different capabilities.
- **New setting:** Nodes able to directly communicate with nodes that cannot communicate back $\rightarrow$ directed graphs.

What consensus algorithms can do for you?



$\mathbf{x} = [1, 0, 2, 1, -1, 1, 2, 2]$

$\mathbf{x}$ represents sensor readings

$\mathbf{y} = [1, 1, 1, 1, 1, 1, 1, 1]$

$\mathbf{y}$ represents target result

distributed control

W Ren, et al. Distributed coordination architecture for multi-robot formation control. RAS 2008.

distributed detection

Jren-Chit Chiny et al. A sensor-cyber network testbed for plume detection, identification, and tracking. IPSN 2007.

synchronization of devices

L. Schenato et al. A distributed consensus protocol for clock synchronization in wireless sensor network. CDC 2007.

distributed estimation

D. Alonso et al. Consensus based distributed estimation of biomass concentration in reverse osmosis membranes. CysWater 2015

maintain data consistency

Paxos algorithm, used by Google.

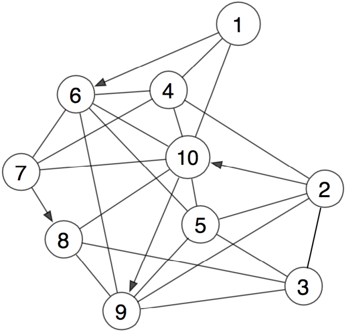
- **General setting:**
  - Network of N nodes with local information ς_i
  - Global objective function $f_0(\varsigma_{\mathbf{1}}, \cdots, \varsigma_{\boldsymbol{N}})$
- **General idea:**
  - Decomposition of main task in separable functions
  - Computed locally by nodes and executed in parallel

$$f_0(\varsigma_{\mathbf{1}}, \cdots, \varsigma_{\boldsymbol{N}}) = \sum_{i=1}^{N} f_i(\varsigma_{\boldsymbol{i}})$$

- **Solution:**

  If each node i takes $x_i(0) = Nf_i(\varsigma_{\boldsymbol{i}})$ as its initial value
  $\rightarrow$ the objective function can be computed as their average.

  Same with product: $f_0(\varsigma_{\mathbf{1}}, \cdots, \varsigma_{\boldsymbol{N}}) = \prod_{i=1}^{N} \gamma_i(\varsigma_{\boldsymbol{i}})$



Example data ($\theta + \mathbf{w}$):

Sensor 1 senses 25°
Sensor 2 senses 26°
Sensor 3 senses 23°
Sensor 4 senses 25°
Sensor 5 senses 28°
Sensor 6 senses 23°
Sensor 7 senses 25°
Sensor 8 senses 21°
Sensor 9 senses 22°
Sensor 10 senses 22°
Average temperature: $\hat{\theta} = 24^\circ$

We want all nodes calculate the average temperature by only using local information

- in a distributed manner
- minimize convergence time
- minimize power consumption

Consensus algorithms:

$+$ simplicity and decentralized philosophy $\rightarrow$ widely used.
- iterative processes $\rightarrow$ repeated exchange of data
- wireless communications $\rightarrow$ high energy consumption.
- sensor nodes on batteries $\rightarrow$ **energy scarce resource**.

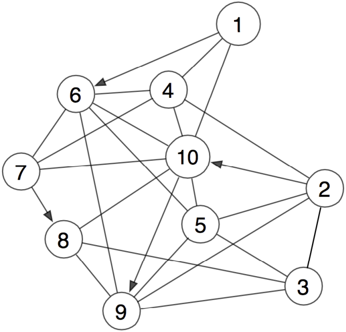
Energy consumption:

power consumption per time step $\times$ number of time steps
(both parameters depend of network topology)

Objective: Topology optimization for minimizing both parameters

Novelty: Directed graphs considered.

Background on Consensus Algorithms



$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Given a graph we can assign
a $N \times N$ adjacency matrix $\mathbf{A}$, given by

$$[\mathbf{A}]_{ij} = \begin{cases} 1 & \text{if exists a link between node i and j} \\ 0 & \text{otherwise} \end{cases}$$

WISENET

UNIVERSITETET I AGDER

- Laplacian $\mathbf{L}$ and Adjacency $\mathbf{A}$ matrix are related as ($\mathbf{L} = \mathbf{D} - \mathbf{A}$):

$$\mathbf{L} = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 7 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

- where $\mathbf{D} = \mathrm{diag}(d_1^{\mathrm{out}}, \dots d_N^{\mathrm{out}})$ is the so-called degree matrix
- and $d_i^{\mathrm{out}} = \sum_{j=1}^{N} [\mathbf{A}]_{ij}$.

- Algebraic connectivity of directed graphs defined as:

$$a(\mathcal{G}) = \min_{\mathbf{x}\in\mathcal{P}} \mathbf{x}^T \mathbf{L}\mathbf{x} = \min_{\mathbf{x}\in\mathbb{R}^N, \mathbf{x}\neq 0, \mathbf{x}\perp\mathbf{e}} \frac{\mathbf{x}^T\mathbf{L}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}$$

  where $\mathcal{P} = \{\mathbf{x} \in \mathbb{R}^N, \mathbf{x} \perp \mathbf{e}, ||\mathbf{x}|| = 1\}$, i.e. set of real vectors of unit norm orthogonal to $\mathbf{e} = [1, \ldots, 1] \in \mathbb{R}^N$.

- For undirected graphs: $a(\mathcal{G}) = \lambda_2(\mathbf{L})$.

- Mirror graph operation $\mathcal{M}(\mathcal{G})$, directed $\mathcal{G} \rightarrow$ undirected $\hat{\mathcal{G}}$:

$$[\hat{\mathbf{A}}]_{ij} = [\hat{\mathbf{A}}]_{ji} = \frac{[\mathbf{A}]_{ij} + [\mathbf{A}]_{ji}}{2},$$

  where $\hat{\mathbf{A}}$ and $\hat{\mathbf{L}} = \hat{\mathbf{D}} - \hat{\mathbf{A}}$ correspond to mirror graph $\hat{\mathcal{G}}$.

- Undo operation $\mathcal{M}^{-1}$, undirected $\hat{\mathcal{G}} \rightarrow$ directed $\mathcal{G}$.

- To maintain topology, each node needs to invest power:

$$\mathbf{p} = (\mathrm{p}_1, \mathrm{p}_2, \ldots, \mathrm{p}_N)$$

where $[\mathbf{p}]_i = \mathrm{p}_i$ denotes the power consumption per communication step of node i.

- Assuming a generic path loss model,
power required by node i to communicate node j is:

$$p_{ij} = p_{\min} r_{ij}^{\gamma}$$

- $p_{\min}$ is minimum power to decode incoming information.
- γ is the path loss exponent.
- r_{ij} is the distance between nodes i and j.

- Let us assume N nodes with initial data $(t = 0)$: $\mathbf{x}(0)$, whose average is: $\mathbf{x}_{\text{avg}} = \frac{\mathbf{1}\mathbf{1}^T\mathbf{x}(0)}{N}$, where $\mathbf{1}$ denotes the all ones column vector.

- General **linear update** of nodes state at time t:

$$\dot{\mathbf{x}}(t) = -\mathbf{L}\mathbf{x}(t)$$

- **Convergence time** $t(\mathbf{L})$ of this linear update is:

$$t(\mathbf{L}) = -t_s \frac{\log(\rho)}{\lambda_2(\hat{\mathbf{L}})}$$

  where $\rho < 1$ is the reduction factor of disagreement between $\mathbf{x}(t)$ and $\mathbf{x}_{\text{avg}}$ and t_s is the duration of a time slot.

- **Graph conditions: strongly connected and balanced.**

- Energy consumption of a node i:

$$\mathcal{E}_i(\mathbf{A}) = \mathcal{E}_i(\hat{\mathbf{A}}) = \mathrm{p}_i \cdot t(\hat{\mathbf{L}}) = \mathcal{K}\, \frac{\sum_{j \in \mathcal{V}} p_{ij} \cdot [\hat{\mathbf{A}}]_{ij}}{\lambda_2(\hat{\mathbf{L}}(\hat{\mathbf{A}}))}$$

- Lifetime of the network:

$$\mathcal{L}(\hat{\mathbf{A}}) = \max_i \left\{ \frac{\mathcal{C}_i}{\mathcal{E}_i(\hat{\mathbf{A}})} \right\}$$

where $\mathcal{C}_i$ denotes the energy budget of node i.

- Meaning: number of consensus processes that can be executed before first node runs out of battery.

Network Lifetime Maximization

- Given maximally connected topology $\mathbf{A}_{\text{max}}$, problem cast:

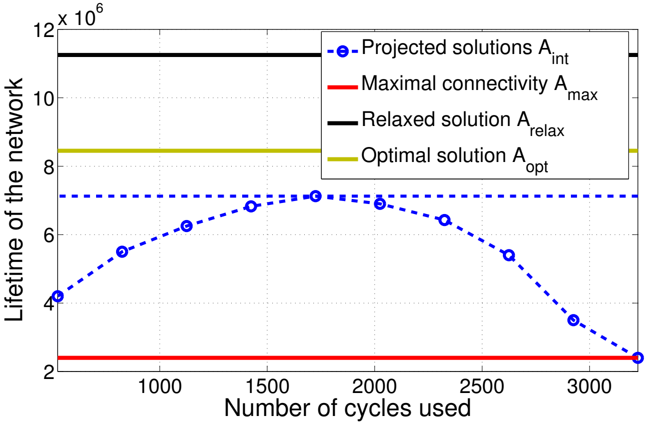
$$
\begin{aligned}
\text{max.}_{\{\hat{\mathbf{A}}\}} \quad & \mathcal{L}(\hat{\mathbf{A}}) \\
\text{s. t.} \quad & \xi \leq \lambda_2(\hat{\mathbf{L}}(\hat{\mathbf{A}})) \\
& [\hat{\mathbf{A}}]_{ij} = [\hat{\mathbf{A}}]_{ji} \ \ \forall i, j \in \mathcal{V} \\
& (\mathcal{M}^{-1}(\mathbf{A}_{\text{max}}, \hat{\mathbf{A}}))\mathbf{1} = (\mathcal{M}^{-1}(\mathbf{A}_{\text{max}}, \hat{\mathbf{A}})^T)\mathbf{1} \\
& [\hat{\mathbf{A}}]_{ij} \in \{0, \tfrac{1}{2}\} \ \text{ if } \ [\mathbf{A}_{\text{max}}]_{ij} \neq [\mathbf{A}_{\text{max}}]_{ji} \\
& [\hat{\mathbf{A}}]_{ij} \in \{0, 1\} \ \text{ if } \ [\mathbf{A}_{\text{max}}]_{ij} = [\mathbf{A}_{\text{max}}]_{ji} = 1 \\
& [\hat{\mathbf{A}}]_{ij} = 0 \ \text{ if } \ [\mathbf{A}_{\text{max}}]_{ij} = [\mathbf{A}_{\text{max}}]_{ji} = 0
\end{aligned}
$$

- ξ small positive constant to ensure graph is connected.
- Second constraint ensures mirror graph is symmetric.
- Third constraint ensures undo graph is balanced.
- Last constraint reduces number of variables.

- After relaxing constraints

$$0 \leq [\hat{\mathbf{A}}]_{ij} \leq \frac{1}{2} \;\text{ if }\; [\mathbf{A}_{\text{max}}]_{ij} \neq [\mathbf{A}_{\text{max}}]_{ji}$$
$$0 \leq [\hat{\mathbf{A}}]_{ij} \leq 1 \;\; [\mathbf{A}_{\text{max}}]_{ij} = [\mathbf{A}_{\text{max}}]_{ji} = 1$$

- Problem becomes a convex-concave fractional problem.
- Solved by using Dinkelbach (parametric) algorithm.
- Relaxed solution obtained $\rightarrow$ projection into feasible set.
- Projection must ensure balanced and connected graph.
- Our projection methodology:
  - Cycles generated according to **A** coefficients.
  - Separated components are connected first and verified.



- Relaxed and projected solutions vs
  true optimal and maximally connected topology.
- Integral gap between relaxed and projected solution shown.

Conclusions and Open Problems

- Topology optimization methodology to maximize network lifetime proposed.
- Solution relies on the notion of mirror graph.
- Solves iteratively convex programs to obtain optimal relaxed solution.
- A novel projection procedure is proposed to recover integer solutions.
- Numerical results showcase benefits of proposed scheme.
- Open problem: Computationally efficient technique for projecting needed.