Synchronization algorithms for Multi-Agent Systems: Analysis, Synthesis and Applications

PhD candidate: Enrico Lovisari

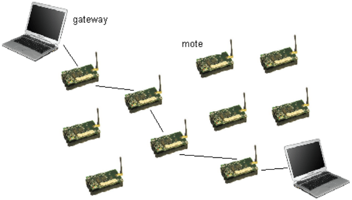
Advisor: prof. Sandro Zampieri
PhD school in Information and Communication technology

Padova, 19/04/2012

Motivation

Growing interest in multi–agent systems

- WSN
-



Motivation

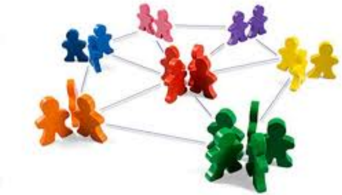
Growing interest in multi–agent systems

- WSN
- robotics
-

Motivation

Growing interest in multi–agent systems

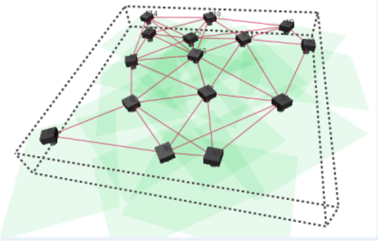
- WSN
- robotics
- opinion dynamics in social networks
-



Motivation

Growing interest in multi–agent systems

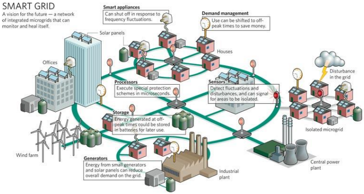
- WSN
- robotics
- opinion dynamics in social networks
- networks of cameras
-



Motivation

Growing interest in multi–agent systems

- WSN
- robotics
- opinion dynamics in social networks
- networks of cameras
- smart grids
-



Motivation

Growing interest in multi–agent systems

- WSN
- robotics
- opinion dynamics in social networks
- networks of cameras
- smart grids
- parallel computing

Motivation

Examples of **global tasks** for Multi–agent systems

WSN	temperature control in buildings
Robots	formation control
Social Networks	information spread/agreement
Camera Networks	patrolling/surveillance of an environment
Smart grids	efficient power production and distribution
Parallel computers	fair distribution of computational load

Motivation

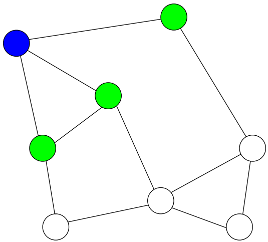
Constraint: agents can only exploit **local information**

- agents can cooperate with a small subset of the network
- no centralized unit

Model: communication graph

$$\mathcal{G} = (\mathcal{V}, \mathcal{E})$$

$\mathcal{V} \rightarrow$ agents
$\mathcal{E} \rightarrow$ allowed communications



Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms

Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms
- Synthesis: criteria for robust synchronization in higher–order consensus networks

Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms
- Synthesis: criteria for robust synchronization in higher–order consensus networks
- Applications:

Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms
- Synthesis: criteria for robust synchronization in higher–order consensus networks
- Applications:
  - Clocks Synchronization

Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms
- Synthesis: criteria for robust synchronization in higher–order consensus networks
- Applications:
  - Clocks Synchronization
  - Cameras calibration in planar networks

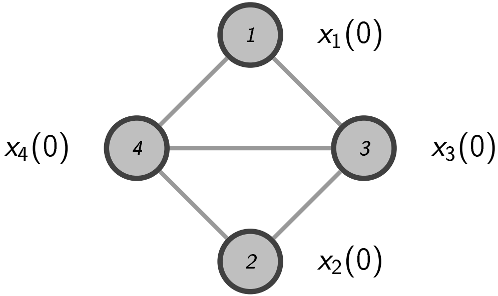
Overview of the Thesis

Synchronization algorithms for Multi–Agent Systems:

- Analysis: performances analysis for consensus algorithms
- Synthesis: criteria for robust synchronization in higher–order consensus networks
- Applications:
  - Clocks Synchronization
  - Cameras calibration in planar networks

Performances indices analysis for consensus algorithms

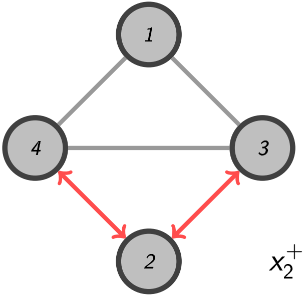
Performances indices analysis



Goal:

- $|x_i - x_j| \xrightarrow{t \to \infty} 0$

Performances indices analysis



$$x_2^+ = f_2(x_2, x_3, x_4)$$

Goal:

- $|x_i - x_j| \xrightarrow{t \to \infty} 0$
- Local communication $\Longrightarrow$ update x_i only using x_j, $j \in \mathcal{N}_i$

Performances indices analysis

Linear consensus

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

- $P = [p_{ij}]$ be associated with the communication graph $\mathcal{G} = (V, \mathcal{E})$

$$p_{ij} > 0 \iff (j, i) \in \mathcal{E}$$

- $P\mathbf{1} = \mathbf{1}$, $\mathcal{G}$ strongly connected

Conclusion:

$$\mathbf{x}(t) \xrightarrow{t\to\infty} \mathbf{1}\pi^T\mathbf{x}(0)$$

where $\pi^T P = \pi^T$.

Performances indices analysis

Performances indices considered

- Rate of convergence: exponential with exponent

$$\rho(P) = \arg\max\{|\lambda|,\ \lambda \neq 1,\ \lambda \in \sigma(\mathrm{P})\}$$

$\longrightarrow$ Goal: fast approach to consensus value

Performances indices analysis

Performances indices considered

- L_2 norm of the trajectory

$$J(P) = \frac{1}{N} \sum_{t \geq 0} \mathbb{E}\left[\|x(t) - x(\infty)\|_2^2\right]$$

$\longrightarrow$ Goal: uniform convergence to consensus value

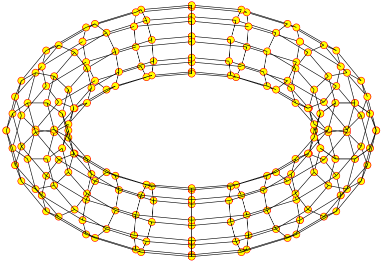
$\longrightarrow$ Asymptotic variance of noisy consensus

$$\mathbf{x}(t+1) = P\mathbf{x}(t) + \mathbf{v}(t)$$

Performances indices analysis

Class of graphs considered:

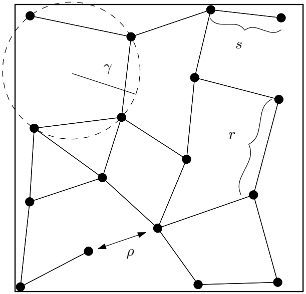
- Cayley graphs: highly structured graphs defined on groups



Performances indices analysis

Class of graphs considered:

- Geometric graphs: agents deployed in $\mathbb{R}^d$ satisfying a set of geometric constraints $\longrightarrow$ "perturbed" Cayley graphs



Barooah, P. and Hespanha, J. *Estimation from relative measurements: Electrical analogy and large graphs*. IEEE Transactions on Signal Processing, 2008

Performances indices analysis

Contribution of Analysis part: generalization to geometric graphs of the following results, already known for Cayley graphs

Performances indices analysis

Contribution of Analysis part: generalization to geometric graphs of the following results, already known for Cayley graphs

Theorem: Let P be a stochastic matrix associated to a geometric graph in dimension d with N agents. Assume that all the nonzero entries of P lie in an interval $[p_{\min}, p_{\max}]$. Then

$$\rho(P) \approx 1 - C\frac{1}{N^{2/d}} \qquad J(P) \approx \begin{cases} N, & d = 1 \\ \log N, & d = 2 \\ 1, & d \geq 3 \end{cases}$$

Performances indices analysis

Conclusions:

Performances of consensus algorithms built on "homogeneous" graphs do not depend, under mild assumptions,

- on the particular graph
- on the particular entries of the matrix P

Moreover, the performances degradation is the same as in the highly structured Cayley graphs

Applications:
synchronization of networks of clocks

Motivation

Linear consensus $\rightarrow$ simple integrators

Motivation

Linear consensus $\rightarrow$ simple integrators

However $\rightarrow$ sometimes too simple model

Motivation

Linear consensus $\rightarrow$ simple integrators

However $\rightarrow$ sometimes too simple model

- Formation control $\rightarrow$ second-order systems

Motivation

Linear consensus → simple integrators

However → sometimes too simple model

- Formation control → second-order systems
- Clock Synchronization → double integrators

Motivation

Linear consensus → simple integrators

However → sometimes too simple model

- Formation control → second-order systems
- Clock Synchronization → double integrators
- Synchronization of generators in power networks → oscillators

Motivation

Linear consensus → simple integrators

However → sometimes too simple model

- Formation control → second-order systems
- Clock Synchronization → double integrators
- Synchronization of generators in power networks → oscillators
- ...

Motivation

Linear consensus → simple integrators

However → sometimes too simple model

- Formation control → second-order systems
- Clock Synchronization → double integrators
- Synchronization of generators in power networks → oscillators
- ...

$\Longrightarrow$ Higher Order Consensus Networks!

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \implies \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \implies \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

Where

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \implies \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

Where

- $N_0 = N_0(z^{-1})$ is any transfer function $\rightarrow$ consensus $N_0(z^{-1}) = \frac{z^{-1}}{1-z^{-1}}$

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \implies \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

Where

- $N_0 = N_0(z^{-1})$ is any transfer function → consensus $N_0(z^{-1}) = \frac{z^{-1}}{1-z^{-1}}$
- Δ_i → perturbation of the nominal N_0

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \Longrightarrow \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

Where

- $N_0 = N_0(z^{-1})$ is any transfer function $\rightarrow$ consensus $N_0(z^{-1}) = \frac{z^{-1}}{1-z^{-1}}$
- $\Delta_i \rightarrow$ perturbation of the nominal N_0
- $L = I - P$ is the Laplacian of a consensus matrix P

(Briefly) Synchronization in HOCN

Synthesis part deals with a generalization of consensus

$$\begin{cases} y_i = N_0(1+\Delta_i)u_i \\ u_i = -\sum_{j\in\mathcal{N}_i} \ell_{ij} y_j \end{cases} \implies \begin{cases} \mathbf{y} = N_0(I+\Delta)u_i \\ \mathbf{u} = -L\mathbf{y} \end{cases}$$

Where

- $N_0 = N_0(z^{-1})$ is any transfer function $\rightarrow$ consensus $N_0(z^{-1}) = \frac{z^{-1}}{1-z^{-1}}$
- $\Delta_i \rightarrow$ perturbation of the nominal N_0
- $L = I - P$ is the Laplacian of a consensus matrix P

(Output–)synchronization:

$$|y_i - y_j| \xrightarrow{t\to\infty} 0$$

(Briefly) Synchronization in HOCN

Contribution of Synthesis part: provides criteria on N_0, Γ and Δ in order to achieve robust output–synchronization

Basic tool: Integral Quadratic Constraint for robust input/output stability

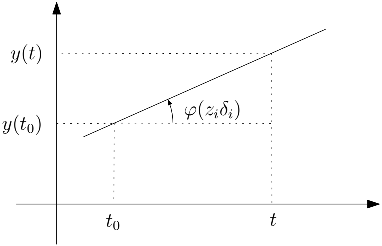
A. Megretski and A. Rantzer, *System analysis via integral quadratic constraints*, TAC, 1997

Algorithms for Clock Synchronization

Model for a clock

- Oscillator $\rightarrow$ event with period δ_i
- Counter $\rightarrow$
  $s_i(t) \approx s_i(t_0) + \delta_i(t - t_0)$
- Time reading

$$y_i(t) = y_i(t_0) + z_i(s_i(t) - s_i(t_0))$$
$$= y_i(t_0) + z_i\delta_i(t - t_0)$$



Algorithms for Clock Synchronization

Assumption: nodes communicate with period $T = 1$ (unrealistic!)

Second order strategy

- node i modifies both y_i and z_i if it is not synchronized with its neighbors

$$\begin{cases} \begin{bmatrix} y_i(t+1) \\ z_i(t+1) \end{bmatrix} = \begin{bmatrix} 1 & \delta_i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_i(t) \\ z_i(t) \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} u_i(t) \\ \mathbf{u}(t) = -L\mathbf{y}(t) \end{cases}$$

Algorithms for Clock Synchronization

Theorem: the network synchronizes if $L = L^T$ and, if $\delta_i = \delta_{\mathrm{nom}} + \tilde{\delta}_i$,

$$\begin{cases} f_1 > 0, f_2 > 0 \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_{\mathrm{nom}}} \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_i} \end{cases} \qquad \lambda_N = \max\{\lambda : \lambda \in \sigma(\Gamma)\}$$

Algorithms for Clock Synchronization

Theorem: the network synchronizes if $L = L^T$ and, if $\delta_i = \delta_{\text{nom}} + \tilde{\delta}_i$,

$$\begin{cases} f_1 > 0, f_2 > 0 \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_{\text{nom}}} \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_i} \end{cases} \qquad \lambda_N = \max\{\lambda : \lambda \in \sigma(\Gamma)\}$$

Moreover exist $\bar{y}$ and $\bar{z}$ such that

$$y_i(t) \xrightarrow{t \to \infty} \bar{z}t + \bar{y}, \forall i$$

Algorithms for Clock Synchronization

Theorem: the network synchronizes if $L = L^T$ and, if $\delta_i = \delta_{\mathrm{nom}} + \tilde{\delta}_i$,

$$\begin{cases} f_1 > 0, f_2 > 0 \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_{\mathrm{nom}}} \\ 0 < \lambda_N < \frac{4}{2f_1 + f_2\delta_i} \end{cases} \qquad \lambda_N = \max\{\lambda : \lambda \in \sigma(\Gamma)\}$$

Moreover exist $\bar{y}$ and $\bar{z}$ such that

$$y_i(t) \xrightarrow{t\to\infty} \bar{z}t + \bar{y}, \forall i$$

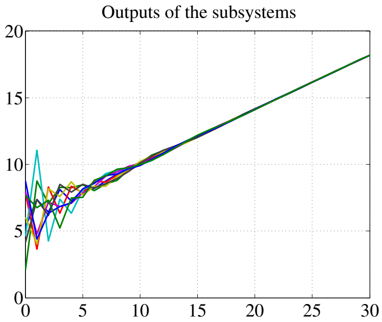
Remark: robustness w.r.t both uncertainty and interconnection!

Algorithms for Clock Synchronization

Corollary: if $\delta_{\text{nom}} = 1$, $f_1 = \frac{1}{2}$, $f_2 = \frac{1}{2T}$, then the network achieves synchronization if

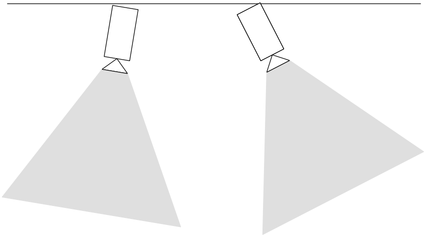
$$0 < \delta_i < 2$$

namely the possible uncertainty is 100% of the nominal value!



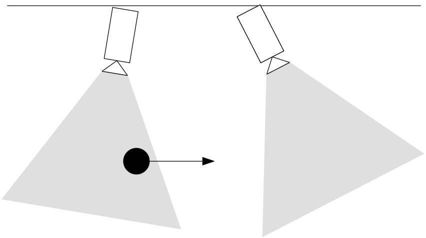
Applications:
calibration in planar networks of cameras

Motivation



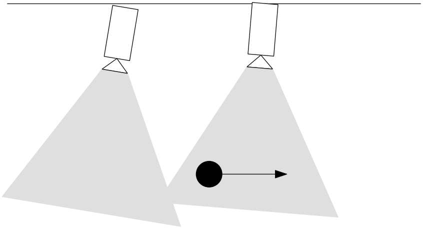
- consider a network of cameras with surveillance tasks

Motivation



- consider a network of cameras with surveillance tasks
- an external agent is exiting from the field of view of a camera

Motivation



- consider a network of cameras with surveillance tasks
- an external agent is exiting from the field of view of a camera
- the cameras must share a common reference frame in order to cooperate and track the agent

Algorithms for Distributed Calibration

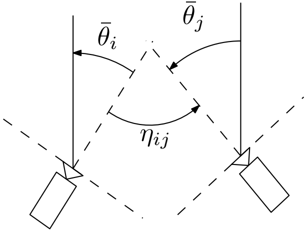
Fix an external reference frame

- Orientation of i-th camera

$$\bar{\theta}_i \in [-\pi, \pi)$$

- Communicating cameras measure in a noisy way their relative orientations

$$\eta_{ij} = (\bar{\theta}_i - \bar{\theta}_j + \varepsilon_{ij})_{2\pi} \in [-\pi, \pi)$$



Algorithms for Distributed Calibration

Goal: given the measurements η_{ij}, compute an estimate $\hat{\theta} \in [-\pi, \pi)^N$ of $\bar{\theta}$ such that

$$\|(\hat{\theta} - \bar{\theta})_{2\pi}\| \quad \text{small}$$

Consequence: the cameras know how they are oriented in the *same* reference frame!

Assumption: $\hat{\theta}_1 = \bar{\theta}_1$, namely 1 is an anchor for the network

Algorithms for Distributed Calibration

First idea: minimize the least–square type cost function

$$V(\theta) = \sum_{(i,j)\in\mathcal{E}} (\theta_i - \theta_j - \eta_{ij})^2_{2\pi}$$

Algorithms for Distributed Calibration

First idea: minimize the least–square type cost function

$$V(\theta) = \sum_{(i,j)\in\mathcal{E}} (\theta_i - \theta_j - \eta_{ij})^2_{2\pi}$$

Remark: with the change of variables $x_i = \theta_i - \bar{\theta}_i$

$$V(\mathbf{x}) = \sum_{(i,j)\in\mathcal{E}} (x_i - x_j - \varepsilon_{ij})^2_{2\pi}$$

- it penalizes the fact that x_i and x_j are different angles...

Algorithms for Distributed Calibration

First idea: minimize the least–square type cost function

$$V(\theta) = \sum_{(i,j)\in\mathcal{E}} (\theta_i - \theta_j - \eta_{ij})^2_{2\pi}$$

Remark: with the change of variables $x_i = \theta_i - \bar{\theta}_i$

$$V(\mathbf{x}) = \sum_{(i,j)\in\mathcal{E}} (x_i - x_j - \varepsilon_{ij})^2_{2\pi}$$

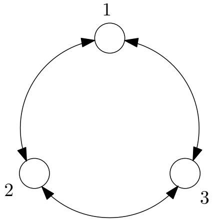
- it penalizes the fact that x_i and x_j are different angles...
- a (noisy) consensus problem on the manifold $\mathcal{S}_1$!

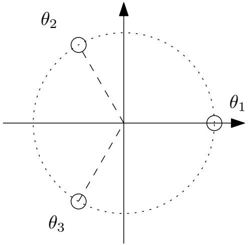
Algorithms for Distributed Calibration

However

- consensus-like algorithm doesn't work due to geometry of $\mathcal{S}_1$
- unacceptable local minima

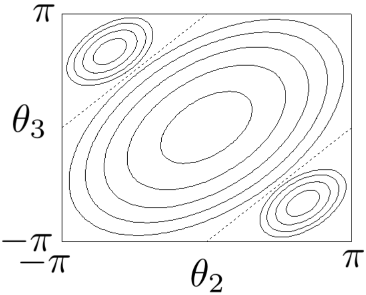
Example: here $\bar{\theta}_i = 0$, $\eta_{ij} = 0$





Algorithms for Distributed Calibration

Fact: $[-\pi, \pi)^N$ is tessellated in regions in which $V(\theta)$ is quadratic



- Each region is associated with a vector of integers $\mathbf{K} \in \mathbb{Z}^M$

Algorithms for Distributed Calibration

Idea: let $\bar{\theta}$ belong to the region associated with $\bar{\mathsf{K}}$

- compute an estimate $\hat{\mathsf{K}}$ of $\bar{\mathsf{K}}$
- minimize the reshaped *quadratic* cost

$$V_{\hat{\mathsf{K}}}(\theta) = \sum_{(i,j)\in\mathcal{E}} (\theta_i - \theta_j - \eta_{ij} - 2\pi K_{ij})^2$$

Algorithms for Distributed Calibration

Estimate of $\bar{K}$

Notice that if γ is cycle $\longrightarrow \sum_{(i,j)\in\gamma} \bar{\theta}_i - \bar{\theta}_j = 0$

Then

$$\sum_{(i,j)\in\gamma} \eta_{ij} = -2\pi \sum_{(i,j)\in\gamma} \bar{K}_{ij} + \sum_{(i,j)\in\gamma} \varepsilon_{ij}$$

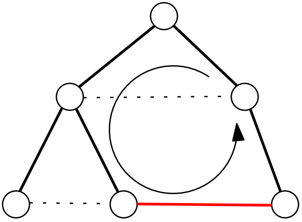
Idea: consider a "sufficiently rich" family Γ of circles (a basis of the *cycle space*) and impose

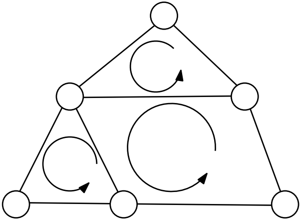
$$\sum_{(i,j)\in\gamma} \hat{K}_{ij} = -\left\lfloor \frac{1}{2\pi} \sum_{(i,j)\in\gamma} \eta_{ij} \right\rceil, \forall\gamma \in \Gamma$$

Algorithms for Distributed Calibration

Two suitable families of cycles

- Fundamental $\mathcal{T}$-cycles: built on a spanning tree
- Minimal cycles: a family of cycles which cannot be "split" in other cycles





Algorithms for Distributed Calibration

Define $G := (A^T A)^\dagger$, A incidence matrix of the graph

Theorem:

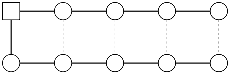
$$\hat{\theta} = \left(\bar{\theta} + \underbrace{G A^T \varepsilon}_{\text{unavoidable}} - 2\pi \underbrace{G A^T (\bar{\mathsf{K}} - \hat{\mathsf{K}})}_{\text{algorithm!}} \right)_{2\pi}$$

Theorem: If $|\varepsilon_{ij}| < \frac{\pi}{L_{\max}}$, then

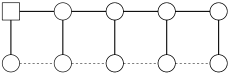
$$\hat{\theta} = \left(\bar{\theta} + G A^T \varepsilon \right)_{2\pi}$$

where $L_{\max}$ is the maximum length of a cycle in Γ

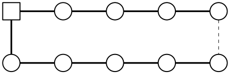
Algorithms for Distributed Calibration



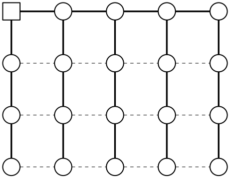
Left: $L_{\mathcal{T}} \sim N,\ L_0 = 4$



Right: $L_{\mathcal{T}} = 4,\ L_0 = 4$



Left: $L_{\mathcal{T}} = N,\ L_0 = N$

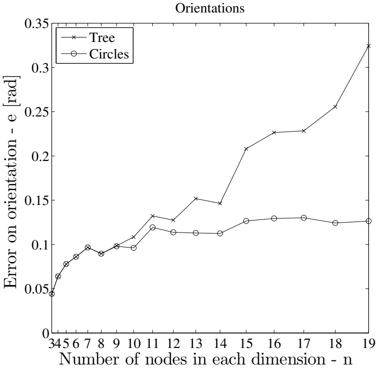


Right: $L_{\mathcal{T}} \sim \sqrt{N},\ L_0 = 4$

Algorithms for Distributed Calibration

For a planar grid we can use a result from the Analysis part:

$$\operatorname{var} GA^T \varepsilon \sim \log N$$



Publications

Journal papers

- E. Lovisari and S.Zampieri, Performance metrics in the average consensus problem: a tutorial. Annual Reviews in Control, Spring '12
- E. Lovisari, F. Garin and S. Zampieri, Performance of the consensus algorithm: a resistance-based approach. SIAM Journal of Optimization and Control (Submitted).

Conference papers

- R. Carli and E. Lovisari, Robust synchronization of networks of heterogeneous double-integrators with applications to wireless sensor networks. 51th IEEE Conference on Decision and Control, CDC'12 (Submitted)
- S. Bolognani, R. Carli, E. Lovisari and S. Zampieri, A randomized linear algorithm for clock synchronization in multi-agent systems. 51th IEEE Conference on Decision and Control, CDC'12 (Submitted)
- D. Borra, E. Lovisari, R. Carli, F. Fagnani and S. Zampieri, Autonomous Calibration Algorithms for Networks of Cameras. American Control Conference, ACC'12.
- E. Lovisari and U. T. Jönsson, A Framework for Robust Synchronization in Heterogeneous Multi-agent Networks. 50th IEEE Conference on Decision and Control, CDC'11
- E. Lovisari, F. Garin and S. Zampieri, A resistance-based approach to consensus algorithm performance analysis. 49th IEEE Conference on Decision and Control, CDC'10
- E. Lovisari and S. Zampieri, Performance metrics in the consensus problem: a Survey. 4th IFAC Symposium on System, Structure and Control, SSSC'10
- E. Lovisari and U. T. Jönsson, A Nyquist criterion for synchronization in networks of heterogeneous linear systems. 2th IFAC Workshop on Distributed Estimation and Control in Networked Systems, NecSys'10
- E. Lovisari, F. Garin and S. Zampieri, A resistance-based approach to performance analysis of the consensus algorithm. 19th International Symposium on Mathematica Theory of Networks and Systems, MTNS'10

Synchronization algorithms for Multi-Agent Systems: Analysis, Synthesis and Applications

PhD candidate: Enrico Lovisari

Advisor: prof. Sandro Zampieri
PhD school in Information and Communication technology

Padova, 19/04/2012