Estimation and control applications of linear consensus algorithms

Luca Schenato
University of Padova

LCCC Workshop, 3-5 February, Lund, Sweden
Networked Control Systems
Group in Padova

Faculty:
- Angelo Cenedese
- Alessandro Chiuso
- Gianluigi Pillonetto
- Luca Schenato
- Sandro Zampieri

PostDocs:
- Federica Garin

Ph.D. students:
- Saverio Bolognani
- Simone Del Favero
- Enrico Lovisari
- Damiano Varagnolo
- Filippo Zanella
Outline

- Consensus algorithms
- Consensus for estimation and control:
  - Distributed estimation
  - Least-square parameter identification
  - Distributed optimization for quadratic cost
  - Sensor calibration
  - Event Detection
  - Time-synchronization
- Experimental results w/ Wireless Sensor Nets
  - RF localization and tracking
  - Time Synchronization
- Control-based metrics for consensus design
Outline

- Consensus algorithms
- Consensus for estimation and control:
  - Distributed estimation
  - Least-square parameter identification
  - Distributed optimization for quadratic cost
  - Sensor calibration
  - Event Detection
  - Time-synchronization
- Experimental results w/ Wireless Sensor Nets
  - RF localization and tracking
  - Time Synchronization
- Control-based metrics for consensus design
Consensus algorithms

- **Main idea**
  - Having a set of agents to agree upon a certain value (usually global function) using only local information exchange (local interaction)

- **Old problem:**
  - Markov Chains (Communications): 60’s
  - Load balancing (Computer Science, Optimization): 80’s (Bertsekas, Tsitsiklis, …
  - Asynchronous iterations (Linear Algebra): 90’s
  - Vehicle Formation Control (Robotics): 90’s (Vicsek, Jadbabaie-Morse, etc …
  - Agreement problem (Economics, signal processing, social networks)
  - Synchronization (Statistical mechanics)
  - ….
Main features

- Distributed computation of general functions

\[ \theta = F(x_1, \ldots, x_N) = f \left( \frac{1}{N} \sum_{i=1}^{N} g_i(x_i) \right) \]

(ex. \( \theta = \frac{1}{N} \sum_{i=1}^{N} x_i \) for \( f = g_i = \text{ident} \))

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

Global Decision Maker
A robotics example: the rendezvous problem

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

If $P$ is doubly stochastic ($1^T P = 1^T$), then $x_i(t) \to \frac{1}{N} \sum_{i=1}^{N} x_i(0)$

**Easy to compute averages of local values (average consensus):**

1) set initial conditions: $x_j(0) = \theta_i$
2) run consensus with doubly stochastic $P$,
3) $x_i(t) \to \frac{1}{N} \sum_{i=1}^{N} \theta_i$
Outline

- Consensus algorithms
- Consensus for estimation and control:
  - Distributed estimation
  - Least-square parameter identification
  - Distributed optimization for quadratic cost
  - Sensor calibration
  - Event Detection
  - Time-synchronization
- Experimental results w/ Wireless Sensor Nets
  - RF localization and tracking
  - Time Synchronization
- Control-based metrics for consensus design
Distributed estimation

\[ y_i = \theta + v_i, \quad v_i \sim \mathcal{N}(0, \sigma_i^2), \quad v_i \perp v_j \]

\[ \hat{\theta}^c = \sum_{i=1}^{N} \alpha_i y_i, \quad \alpha_i = \frac{1/\sigma_i^2}{\sum_{j=1}^{N} 1/\sigma_j^2} \]

\[ \hat{\theta}^c = \frac{\sum_{i=1}^{N} y_i/\sigma_i^2}{\sum_{i=1}^{N} 1/\sigma_i^2} = \frac{1}{N} \sum_{i=1}^{N} \frac{y_i/\sigma_i^2}{1/\sigma_i^2} \]

Strategy:

\[ x_i^y(0) = y_i/\sigma_i^2, \quad x_i^\sigma(0) = 1/\sigma_i^2 \]

run two average consensus in parallel on \( x_i^y \) and \( x_i^\sigma \) so that

\[ x_i^y(t) \to \frac{1}{N} \sum_{i=1}^{N} y_i/\sigma_i^2, \quad x_i^\sigma(t) \to \frac{1}{N} \sum_{i=1}^{N} 1/\sigma_i^2 \]

therefore

\[ \tilde{\theta}_i(t) = \frac{x_i^y(t)}{x_i^\sigma(t)} \to \hat{\theta}^c \]
Least-square identification

Estimate

\[ f(x) = \sum_{m=1}^{M} \theta_m f_m(x) \]

with unknown parameters \( \theta_1, \ldots, \theta_M \) from noisy measurements

\[ y_i = \sum_{m=1}^{M} \theta_m f_m(x_i) + v_i, \quad i = 1, \ldots, N \]

By stacking all measurements

\[
\begin{bmatrix}
  y(x_1) \\
  y(x_2) \\
    \vdots \\
  y(x_N)
\end{bmatrix} =
\begin{bmatrix}
  f_1(x_1) & \cdots & f_M(x_1) \\
  f_1(x_2) & \cdots & f_M(x_2) \\
    \vdots & \ddots & \vdots \\
  f_1(x_N) & \cdots & f_M(x_N)
\end{bmatrix}
\begin{bmatrix}
  \theta_1 \\
  \vdots \\
  \theta_M
\end{bmatrix}
+ 
\begin{bmatrix}
  v_1 \\
  \vdots \\
  v_N
\end{bmatrix}
\]

or equivalently:

\[ y = F\theta + v \]

Goal:

\[ \hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{N} v_i^2 = \arg\min_{\theta} ||F\theta - b||^2 = (F^T F)^{-1} F^T y \]

can be written as

\[ \hat{\theta} = (\sum_{i=1}^{N} F_i F_i^T)^{-1} (\sum_{i=1}^{N} F_i y_i) = \left( \frac{1}{N} \sum_{i=1}^{N} F_i F_i^T \right)^{-1} \left( \frac{1}{N} \sum_{i=1}^{N} F_i y_i \right) \]

(Xiao,Boyd,Lall, IPSN05), (Bolognani,Del Favero, Schenato, Varagnolo JRNC10)
Distributed quadratic optimization

\[ f_i(x): \text{local cost function (convex)} \]

\[ J(x) = \sum_{i=1}^{N} f_i(x): \text{global cost function} \]

\[ \min_{x} J(x) = \sum_{i=1}^{N} f_i(x) \text{ (convex)} \]

\[ \min_{x_1, \ldots, x_N} \sum_{i=1}^{N} f_i(x_i) \]

\[ \text{s.t. } x_i = x_j \text{ for } (i, j) \text{ in comm. graph} \]

\[ f_i(x) = x^T S_i x - 2x^T b_i + c_i: \text{quadratic cost function} \]

then \[ J(x) = x^T (\sum_i S_i) x - 2x^T (\sum_i b_i) + (\sum_i c_i) \]

\[ x^* = \left( \frac{1}{N} \sum_i S_i \right)^{-1} \left( \frac{1}{N} \sum_i b_i \right) \]
Sensor Calibration

\[ P^{ij} = g(\xi_i, \xi_j) + o_i \]
\[ P^{ji} = g(\xi_j, \xi_i) + o_j \]

\( g() \) unknown but symmetric, i.e. \( g(\xi_i, \xi_j) = g(\xi_j, \xi_i) \), then \( P^{ij} - P^{ji} = o_i - o_j \)

Design \( \hat{o}_i \) so that \( o_i - \hat{o}_i = 0 \): impossible

Design \( \hat{o}_i \) so that \( o_i - \hat{o}_i = \alpha \), \( \alpha \) small: easy

Strategy:
1) set \( x_j = o_i - \hat{o}_i \) write consensus for \( x_i \)
2) \( \hat{o}_i(t + 1) = \hat{o}_i(t) - \sum_{j \in \mathcal{N}_i} p_{ij} \left( P^{ij} - P^{ji} - \hat{o}_i(t) + \hat{o}_j(t) \right) \)
3) \( \hat{o}_i(t) \rightarrow o_i - \frac{1}{N} \sum_i o_i = o_i - \alpha \approx o_i \)

(Bolognani, Del Favero, Schenato, Varagnolo JRNC10)
Event detection

We have a binary random variable $x$ such with prior

$$P(x = 0) = P(x = 1) = 1/2$$

$N$ sensors can estimate $x$ though a binary random variable $y_i$ which are conditional independent and with conditional probabilities

$$P(y = 1|x = 0) = P(y = 0|x = 1) = e_i$$

$$P(y = 0|x = 0) = P(y = 1|x = 1) = 1 - e_i$$

It can be seen that the normalized log-likelihood function is

$$\mathcal{L}(y_1, \ldots, y_N) = \frac{1}{N} \log \frac{P(0|y_1, \ldots, y_N)}{P(1|y_1, \ldots, y_N)} = \frac{1}{N} \sum_i (1-2y_i) \log \frac{1-e_i}{e_i}$$

$$\hat{x} = 0 \iff \mathcal{L}(y_1, \ldots, y_N) > 0$$

Strategy:

$$x_i(0) = (1-2y_i) \log \frac{1-e_i}{e_i}$$

run average consensus $x_i(t)$ so that

$$x_i(t) \rightarrow \mathcal{L}(y_1, \ldots, y_N)$$
**Time Synchronization**

Local clocks
\[ \tau_i(t) = \alpha_i t + \beta_i \quad i = 1, \ldots, N \]

Virtual reference clock
\[ \tau^*(t) = \alpha^* t + \beta^* \]

Local clock estimate
\[ \hat{\tau}_j(t) = \hat{\alpha}_j \tau_i + \hat{o}_j \quad i = 1, \ldots, N \]

\[ \hat{\tau}_j(t) = \hat{\alpha}_j \alpha_j t + \hat{\alpha}_i \beta_i + \hat{o}_j \]

GOAL: find \((\hat{\alpha}_j, \hat{o}_j)\) such that
\[ \lim_{t \to \infty} \hat{\tau}_i(t) = \tau^*(t), \forall i = 1, \ldots, N \]

**Strategy:**
1) set \(x_j^\alpha = \alpha_j \hat{\alpha}_j\) and \(x_j^\beta = \hat{o}_j + \hat{\alpha}_j \beta_j\) write consensus
2) find update equations for \(\hat{\alpha}_j(t)\) and \(\hat{o}_j(t)\)
3) \(\alpha_i \hat{\alpha}_i(t) \to \frac{1}{N} \sum_{i=1}^{N} \alpha_i\) and \(\hat{o}_j(t) + \hat{\alpha}_j(t) \beta_j \to \beta^*\)

(Solis, Borkar, Kumar, CDC06, Gamba, Schenato, CDC07 Carli, Chiuso, Schenato, Zampieri, IFAC08)
Outline

- Consensus algorithms
- Consensus for estimation and control:
  - Distributed estimation
  - Least-square parameter identification
  - Distributed optimization for quadratic cost
  - Sensor calibration
  - Event Detection
  - Time-synchronization
- Experimental results w/ Wireless Sensor Nets
  - RF localization and tracking
  - Time Synchronization
- Control-based metrics for consensus design
Sensor calibration

\[ \Delta \bar{P}^{ij} = \bar{P}^{ij} - \bar{p}^{ij} = o_i - o_j \]

**Error distribution:**
before ( ) and after ( )

<table>
<thead>
<tr>
<th>Error Range</th>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;0.5 dB</td>
<td>24%</td>
<td>56%</td>
</tr>
<tr>
<td>&lt;1</td>
<td>50%</td>
<td>88%</td>
</tr>
<tr>
<td>&gt;2dB</td>
<td>35%</td>
<td>0.6%</td>
</tr>
<tr>
<td>Max</td>
<td>&lt;6dB</td>
<td>&lt;3.5dB</td>
</tr>
</tbody>
</table>

(Bolognani, Del Favero, Schenato, Varagnolo JRNC10)
Model identification

\[ P^{ji} = f(d_{ij}, \theta) = \frac{e^{\theta_1}}{\|d_{ij}\|^{\theta_2}} + \text{noise}, \quad \theta \text{ unknown parameters} \]

\[ \log(P^{ji}) = \theta_1 - \log(d_{ij}) \theta_2 + \text{noise}, \quad P^{ij}, d_{ij} \text{ known parameters} \]

\[ y = F\theta + \nu \]

(Bolognani, Del Favero, Schenato, Varagnolo JRNC10)
Time Synch for WSNs

Tmote Sky nodes

7x5 grid (10 hops)

(Fiorentin, Schenato, Necsys09)

Error vs distance

Error vs time
Outline

- Consensus algorithms
- Consensus for estimation and control:
  - Distributed estimation
  - Least-square parameter identification
  - Distributed optimization for quadratic cost
  - Sensor calibration
  - Event Detection
  - Time-synchronization
- Experimental results w/ Wireless Sensor Nets
  - RF localization and tracking
  - Time Synchronization
- Control-based metrics for consensus design
How to design consensus?

Consensus algorithm:

\[ x(t+1) = Px(t), \] \( P \) consistent with comm graph \( G \)

how to design \( P \)?

**Stability condition:** if \( P \) stochastic then equivalent to connectivity of \( G_P \)

**Stability design:** Metropolis weights, Gossip, Broadcast (distributed)

**Performance metrics:**
- Rate of convergence: \( |\lambda_2(P)| \) **Well studied**
\[ y_i = \theta + v_i, \quad v_i \sim \mathcal{N}(0, 1), \quad \hat{\theta}^c = \frac{1}{N} \sum_i y_i, \quad \text{Var}(\theta - \hat{\theta}^c) = \frac{1}{N} \]

\[ x(t + 1) = Px(t), \quad x(0) = [y_1 \ y_2 \ \ldots \ y_N]^T \]

If \( P \) only stochastic \( \lim_{t \to \infty} x(t) = \hat{\theta}1 \), \( \text{Var}(\theta - \hat{\theta}) = \|\rho\|_2 \),
where \( \rho \) left eigenvector of \( P \) for eigenvalue 1 \( (\frac{1}{N} \leq \|\rho\|_2 \leq 1) \).

If \( P \) doubly stochastic and normal \( (PP^T = P^TP) \),
then \( \frac{1}{N} \sum_i \text{Var}(\theta - x_i(t)) = \frac{1}{N} \sum_{\lambda_j \in \Lambda(P)} |\lambda_j|^2 \),
convex

(Carli, Garin Zampieri, ITA09)
Noisy consensus

\[ x(t + 1) = Px(t) + v(t), \quad v(t) \sim \mathcal{N}(0, I) \]

\[ \bar{x}(t) = \frac{1}{N} \sum_i x_i(t) \text{ instantaneous average,} \]

P doubly stochastic and normal, then

\[
\lim_{t \to \infty} \mathbb{E}[\|x(t) - \bar{x}(t)1\|_2] = \frac{1}{N} \sum_{\lambda_j \in \Lambda(P), \lambda \neq 1} \frac{1}{1 - |\lambda_j|^2}
\]

convex

(Xiao, Boyd, Kim, JPDC07)
Control-based performance metrics

- Distance from consensus: \( \|\rho\|_2 \)

- \( L_2 \) performance: \( \sum_{\lambda_i \neq 1} \frac{1}{1 - |\lambda_i|^2} \)

- Estimation performance: \( \sum_{\lambda_i} |\lambda_i|^{2t} \)

- Consensus-based Kalman Filter \( J = \sum_{\lambda_i \neq 1} \frac{|\lambda_i|^{2t}}{1 - (1-l)^2 |\lambda_i|^{2t}} \) 
  (Carli, Chiuso, Schenato, Zampieri, JSAC08)

- Consensus-based Time-synch \( J = \sum_{\lambda_i} f_i(\lambda_i) \), 
  \( f_i \) convex 
  (Carli, Chiuso, Schenato, Zampieri, IFAC08)

- …..
Example

\[ x(t + 1) = Px(t), \ P \ \text{consistent w/ graph} \]

rate of convergence:
\[ \lambda_2 \approx 1 - \frac{8}{N^2} \] (very bad!!)

estimation performance:
\[ \frac{1}{N} \sum_i \text{Var}(x_i(t) - \theta) \leq \frac{3}{N}, \forall t \geq 1 \] (almost optimal!!!)

(Boyd, Diaconis, Parrillo, Xiao, IM07)
Summary

- Consensus is good for quadratic problems
- Approximate non-quadratic local costs to apply consensus:
  - Camera networks calibration
  - Smart power grids control
- Linear consensus vs Lagrange-based distributed optimization
- Control performance metrics provide new twist to the “old” consensus problem
THANK YOU