Distributed Parametric-Nonparametric Estimation in Networked Control Systems

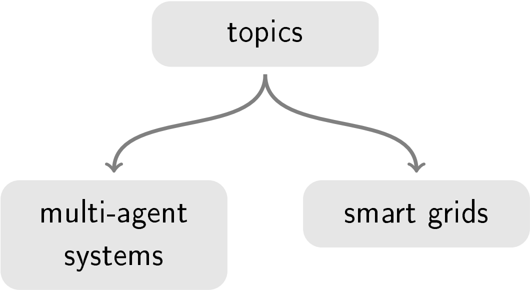
Damiano Varagnolo

Department of Information Engineering - University of Padova

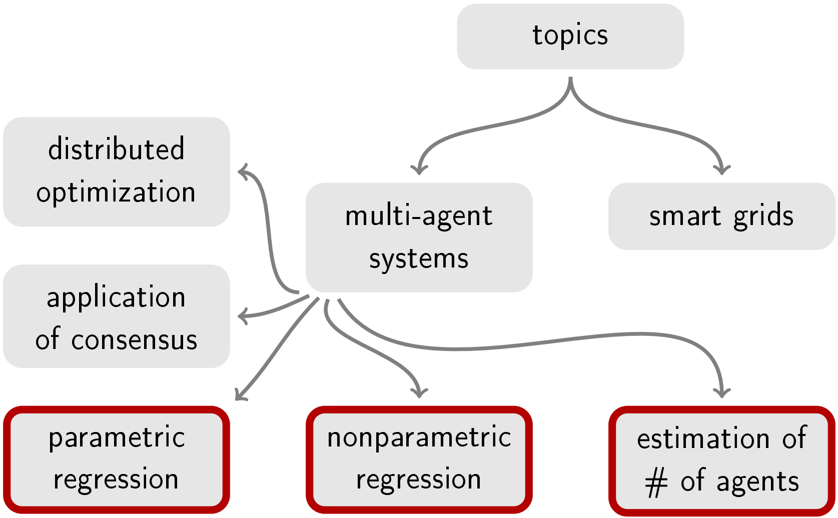
April, 18^{th} 2011

entirely written in LaTeX2ε using Beamer and TikZ

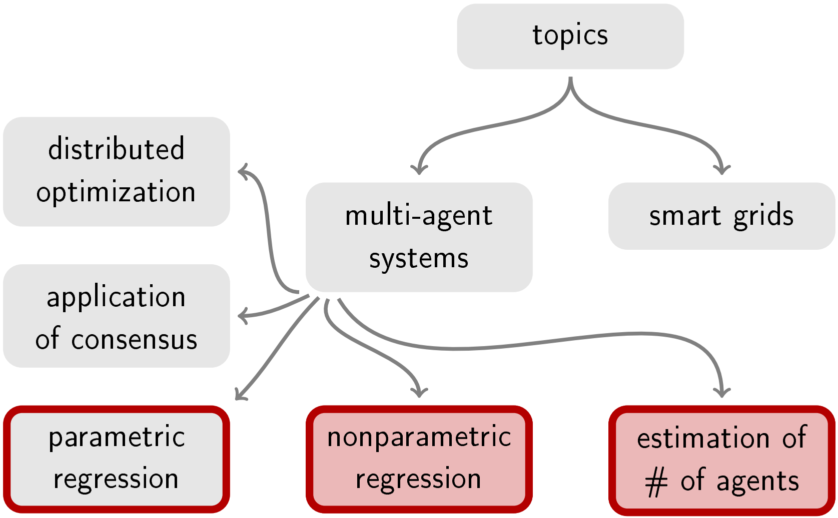
Research topics



Research topics



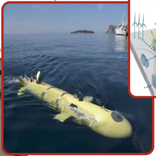
Research topics



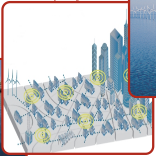
Multi-agent systems: examples of applications

Multi-Robot Coordination

Underwater Exploration



Smart Grids Deployment



Wind Power Management

Wild Life Monitoring

Smart Highways Deployment

Smart Surveillance Implementation

First problem considered in this speech

Assumption

noisy measurements of

$$f(x,t) \ : \ \mathbb{R}^3 \times \mathbb{R} \mapsto \mathbb{R}$$

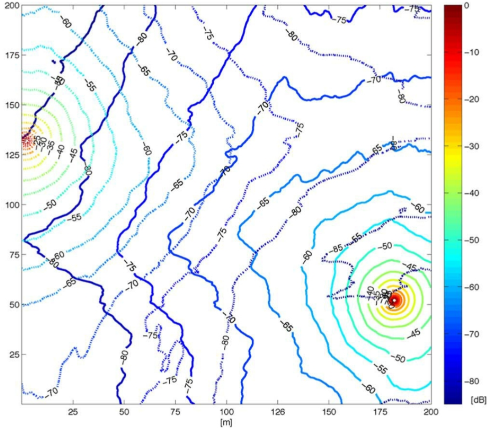
that are

- non uniformly sampled in space x
- non uniformly sampled in time t
- taken by different agents

Objective

smoothing in space (x) and forecast in time (t) the quantity $f(x,t)$

Example 1 - channel gains in geographical areas



source: Dall'Anese et al., 2011

$x \in \mathbb{R}^2$: position

t : time

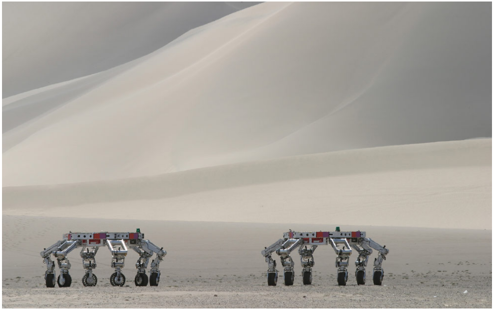
$f(x, t)$: channel gain

Example 2 - waves power extraction

source: www.graysharboroceanenergy.com

$$\begin{aligned} x \in \mathbb{R}^2 &: \quad \text{position} \\ t &: \quad \text{time} \\ f(x,t) &: \quad \text{sea level} \end{aligned}$$

Example 3 - multi robot exploration



source: http://www-robotics.jpl.nasa.gov

$x \in \mathbb{R}^2$: position

$f(x)$: ground level

Difficulties related to this problem

Information-related difficulties

- non-uniform samplings both in time and in space
- unknown dynamics of f
- unknown or extremely complex correlations in time and space

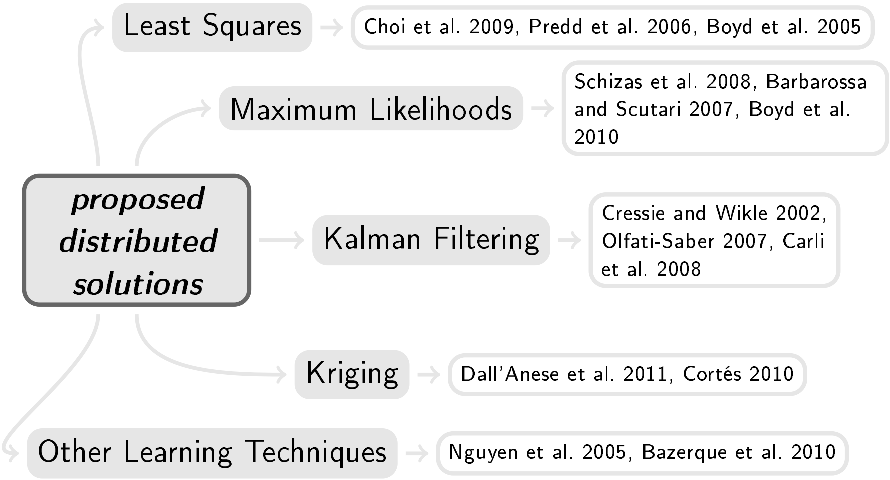
Hardware-related difficulties

- energy & computational & memory & bandwidth limitations

Framework-related difficulties

- mobile and time varying network

State of the art



State of the art

dynamic scenarios

proposed distributed solutions

- Least Squares → Choi et al. 2009, Predd et al. 2006, Boyd et al. 2005
- Maximum Likelihoods → Schizas et al. 2008, Barbarossa and Scutari 2007, Boyd et al. 2010
- Kalman Filtering → Cressie and Wikle 2002, Olfati-Saber 2007, Carli et al. 2008
- Kriging → Dall'Anese et al. 2011, Cortés 2010
- Other Learning Techniques → Nguyen et al. 2005, Bazerque et al. 2010

State of the art

static scenarios

proposed distributed solutions

- **Least Squares** → Choi et al. 2009, Predd et al. 2006, Boyd et al. 2005
- **Maximum Likelihoods** → Schizas et al. 2008, Barbarossa and Scutari 2007, Boyd et al. 2010
- Kalman Filtering → Cressie and Wikle 2002, Olfati-Saber 2007, Carli et al. 2008
- **Kriging** → Dall'Anese et al. 2011, Cortés 2010
- **Other Learning Techniques** → Nguyen et al. 2005, Bazerque et al. 2010

State of the art - Vision

obtain

$$\widehat{f}(x,t) = \Psi\,(\text{past measurements})$$

being

- distributed
- capable of both smoothing and prediction

Our approach

nonparametric: $\Psi\,(\cdot)$ lives in an ***infinite dimensional space***

Why should we use a nonparametric approach?

Motivations

- it could be difficult or even impossible to define a parametric model (e.g. when only regularity assumptions are available)
- parametric models could involve a large number of parameters (could require nonlinear optimization techniques)
- lead to convex optimization problems
- consistent, i.e. $\widehat{f} \rightarrow f$ when $\#$ measurements $\rightarrow \infty$ (De Nicolao, Ferrari-Trecate, 1999)

State of the art - where we actually contributed

our small puzzle-piece

agents estimate the same f

e.g. exploration

static scenario (f independent of t)

no needs to discard old measurements

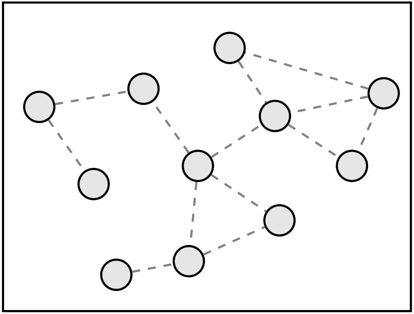
Regularization Network approach

Poggio and Girosi 1990

Our goal

obtain a *simple*, *self-evaluating* and *auto-tuning* multi-agent regression strategy

Framework



Agents:

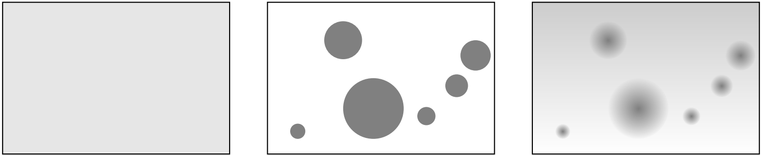
- noisily sample the same f
- limited computational & communication capabilities
- 1 measurement $\times$ agent (ease of notation)
- M measurements in total

Measurement model

$$y_m = f(x_m) + \nu_m \qquad (1)$$

- $f : \mathcal{X} \subset \mathbb{R}^d \to \mathbb{R}$ unknown ($\mathcal{X}$ compact)
- $\nu_m \perp x_m$, zero mean and variance σ^2
- $x_m \sim \mu$ i.i.d. (*agents know μ!!*)

examples of μ:



uniform jitter generic

Considered cost function

$$Q(f) = \sum_{m=1}^{M} \left(y_m - f(x_m)\right)^2 + \gamma \|f\|_K^2$$

Centralized optimal solution as a Regularization Network

$$f_c = \sum_{m=1}^{M} c_m K(x_m, \cdot) \qquad \begin{bmatrix} c_1 \\ \vdots \\ c_M \end{bmatrix} = \left(\begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_M) \\ \vdots & & \vdots \\ K(x_M, x_1) & \cdots & K(x_M, x_M) \end{bmatrix} + \gamma I \right)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}$$

Considered cost function

$$Q(f) = \sum_{m=1}^{M} \left(y_m - f(x_m)\right)^2 + \gamma \|f\|_K^2$$

$Q(f)$: lives in an infinite dimensional space

$\gamma \|f\|_K^2$: regularization factor, $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ Mercer kernel

Centralized optimal solution as a Regularization Network

$$f_c = \sum_{m=1}^{M} c_m K(x_m, \cdot) \qquad \begin{bmatrix} c_1 \\ \vdots \\ c_M \end{bmatrix} = \left(\begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_M) \\ \vdots & & \vdots \\ K(x_M, x_1) & \cdots & K(x_M, x_M) \end{bmatrix} + \gamma I \right)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}$$

Drawbacks

$$f_c = \sum_{m=1}^{M} c_m K(x_m, \cdot) \qquad \begin{bmatrix} c_1 \\ \vdots \\ c_M \end{bmatrix} = \left(\begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_M) \\ \vdots & & \vdots \\ K(x_M, x_1) & \cdots & K(x_M, x_M) \end{bmatrix} + \gamma I \right)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}$$

- computational cost: $O(M^3)$ (inversion of $M \times M$ matrix)
- transmission cost: $O(M)$ (knowledge of whole $\{x_m, y_m\}_{m=1}^{M}$)

$$\Downarrow$$

need to find alternative solutions

Alternative centralized optimal solution (1^{st} on 2)

Structure of K implies

- $K(x_1, x_2) = \sum_{e=1}^{+\infty} \lambda_e \phi_e(x_1) \phi_e(x_2)$
- $f(x) = \sum_{e=1}^{+\infty} b_e \phi_e(x)$

λ_e = eigenvalue
ϕ_e = eigenfunction

$\Rightarrow$ measurement model can be rewritten as

$$y_m = \overbrace{\left[\phi_1(x_m), \phi_2(x_m), \ldots\right]}^{C_m :=} \overbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix}}^{b :=} + \nu_m \qquad (2)$$

Alternative centralized optimal solution (2^{nd} on 2)

$$b_c = \left(\frac{1}{M} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{M} \sum_{m=1}^{M} C_m^T C_m \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} C_m^T y_m \right) \qquad (3)$$

involves infinite dimensional objects:

$$b_c = \begin{bmatrix} \bullet & \cdots & \cdots \\ \vdots & \ddots & \\ \vdots & & \ddots \end{bmatrix}^{-1} \begin{bmatrix} \bullet \\ \vdots \\ \vdots \end{bmatrix}$$

$\Rightarrow$ ***cannot be computed exactly***

Suboptimal finite dimensional solution

New estimator

$$b_r = \left(\frac{1}{M} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T C_m^E \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T y_m \right)$$

- computable (involves $E \times E$ matrices and E-dimensional vectors)
- minimizes $Q^E(b) := \sum_{m=1}^{M} \left(y_m - C_m^E b \right)^2 + \gamma \sum_{e=1}^{E} \frac{b_e^2}{\lambda_e}$

Suboptimal finite dimensional solution

New estimator

$$b_r = \left(\frac{1}{M} \mathrm{diag} \left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T C_m^E \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T y_m \right)$$

- computable (involves $E \times E$ matrices and E-dimensional vectors)
- minimizes $Q^E(b) := \sum_{m=1}^{M} \left(y_m - C_m^E b \right)^2 + \gamma \sum_{e=1}^{E} \frac{b_e^2}{\lambda_e}$

Drawbacks

1. $O\left(E^3\right)$ computational effort
2. $O\left(E^2\right)$ transmission effort
3. must know M

Derivation of the distributed estimator

$$b_r = \left(\frac{1}{M} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T C_m^E \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T y_m \right)$$

Consider the approximations

- $M \to M_g$ (guess)
- $\frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T C_m^E \to \mathbb{E}_\mu \left[\left(C_m^E \right)^T C_m^E \right] = I$

Derivation of the distributed estimator

obtain:

$$b_d = \left(\frac{1}{M_g} \mathrm{diag} \left(\frac{\gamma}{\lambda_e} \right) + I \right)^{-1} \left(\frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T y_m \right)$$

Advantages

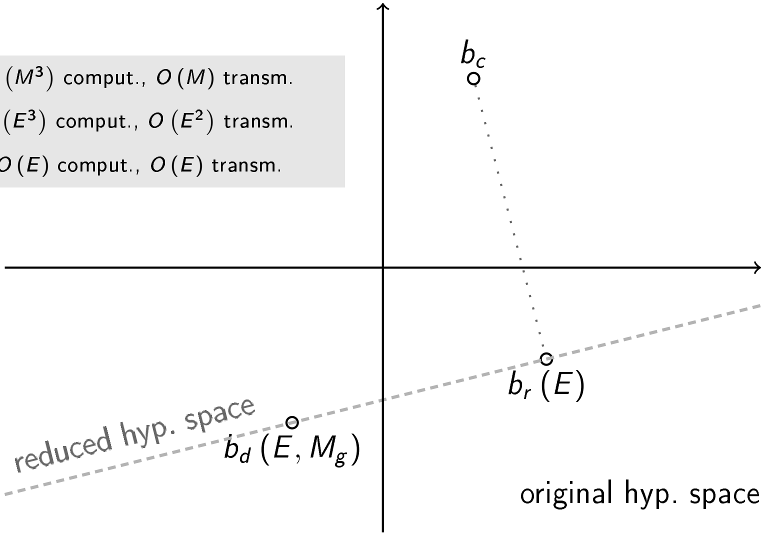
1. $O(E)$ computational effort
2. $O(E)$ transmission effort

Summary of proposed estimation schemes

b_c: $O(M^3)$ comput., $O(M)$ transm.

b_r: $O(E^3)$ comput., $O(E^2)$ transm.

b_d: $O(E)$ comput., $O(E)$ transm.

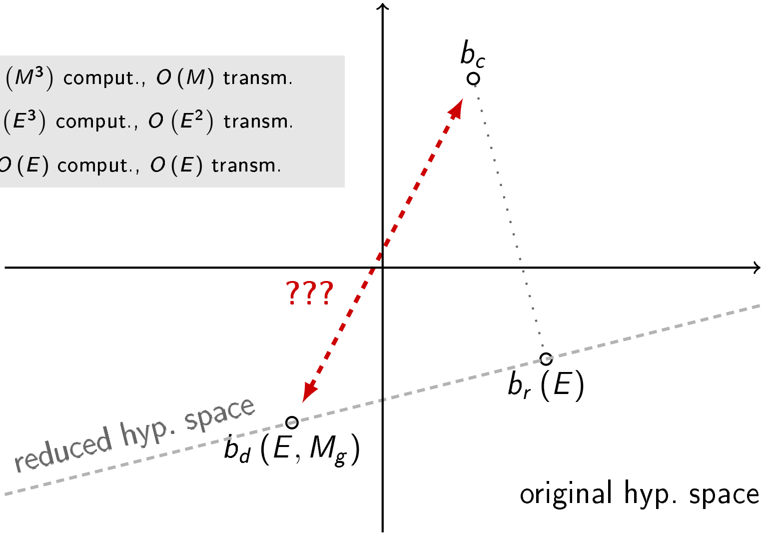


Summary of proposed estimation schemes

b_c: $O(M^3)$ comput., $O(M)$ transm.

b_r: $O(E^3)$ comput., $O(E^2)$ transm.

b_d: $O(E)$ comput., $O(E)$ transm.

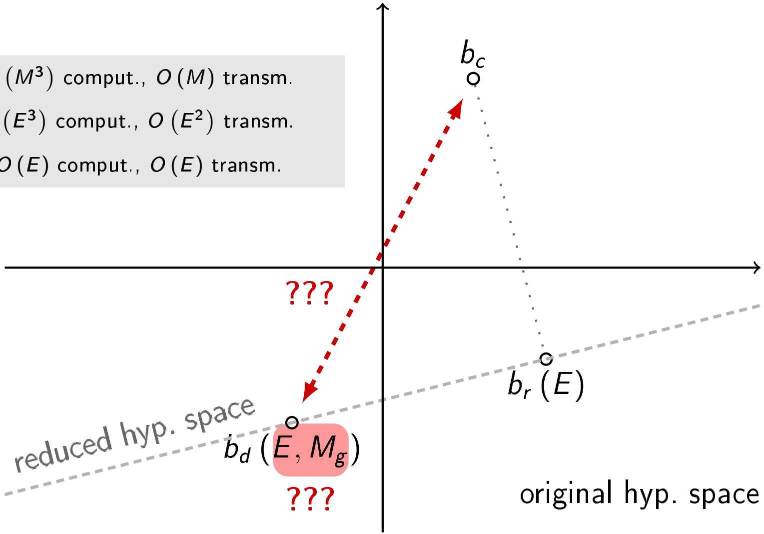


Summary of proposed estimation schemes

b_c: $O(M^3)$ comput., $O(M)$ transm.

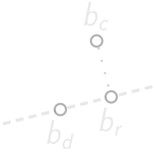
b_r: $O(E^3)$ comput., $O(E^2)$ transm.

b_d: $O(E)$ comput., $O(E)$ transm.



Quantification of performances

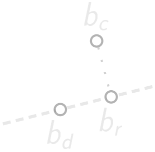
Assumption: E, M_g already chosen, b_d already computed



$$\|b_c - b_d\|_2 \;\leq\; \frac{1}{M}\sum_{m=1}^{M} |r_m| \;+\; \|U_M b_d\|_2 \;+\; \|U_C b_d\|_2$$

Quantification of performances

Assumption: E, M_g already chosen, b_d already computed

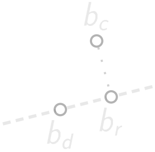


$$\|b_c - b_d\|_2 \leq \frac{1}{M}\sum_{m=1}^{M} |r_m| + \|U_M b_d\|_2 + \|U_C b_d\|_2$$

local residuals

Quantification of performances

Assumption: E, M_g already chosen, b_d already computed



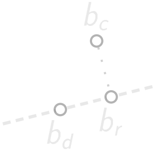
$$\propto \frac{1}{M_{\min}} - \frac{1}{M_{\max}}$$

$$\left\| b_c - b_d \right\|_2 \;\leq\; \frac{1}{M}\sum_{m=1}^{M} |r_m| \;+\; \left\| U_M b_d \right\|_2 \;+\; \left\| U_C b_d \right\|_2$$

local residuals

Quantification of performances

Assumption: E, M_g already chosen, b_d already computed



$$\propto \frac{1}{M_{\min}} - \frac{1}{M_{\max}}$$

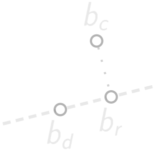
$$\|b_c - b_d\|_2 \leq \frac{1}{M}\sum_{m=1}^{M} |r_m| + \|U_M b_d\|_2 + \|U_C b_d\|_2$$

local residuals

$$\propto I - \frac{1}{M}\sum_{m=1}^{M} \left(C_m^E\right)^T C_m^E$$

Quantification of performances

Assumption: E, M_g already chosen, b_d already computed



$$\propto \frac{1}{M_{\min}} - \frac{1}{M_{\max}}$$

$$\left\| b_c - b_d \right\|_2 \leq \frac{1}{M} \sum_{m=1}^{M} |r_m| + \left\| U_M b_d \right\|_2 + \left\| U_C b_d \right\|_2$$

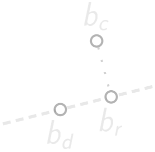
local residuals

$$\propto I - \frac{1}{M} \sum_{m=1}^{M} \left(C_m^E \right)^T C_m^E$$

computable through distributed MC

Tuning of the parameters - key ideas

Assumption: have some information on the energy of f

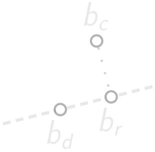


parameters to be estimated

- number of eigenfunctions E
- number of measurements M_g

Tuning of the parameters - key ideas

Assumption: have some information on the energy of f



parameters to be estimated

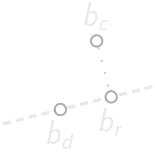
number of eigenfunctions E

number of measurements M_g

assure $\left\| b_c - b_r(E) \right\|$ to be sufficiently small

Tuning of the parameters - key ideas

Assumption: have some information on the energy of f

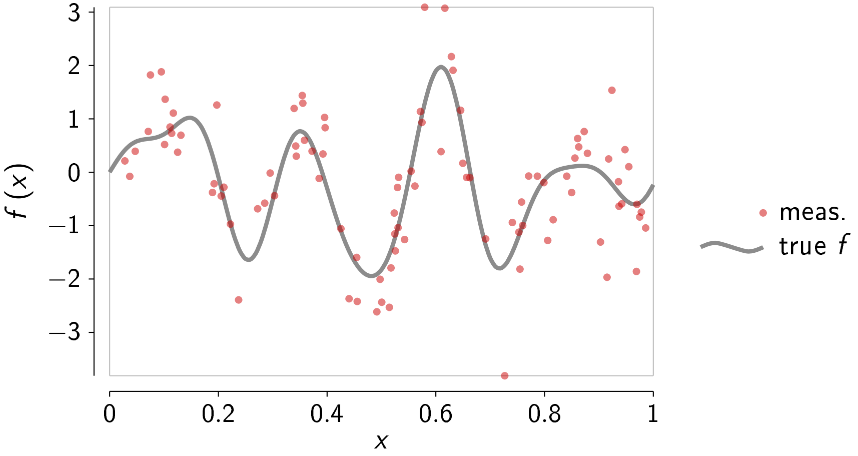


parameters to be estimated

- number of eigenfunctions E
  - assure $\|b_c - b_r(E)\|$ to be sufficiently small
- number of measurements M_g
  - minimize the bound on $\|b_c - b_d(E, M_g)\|$

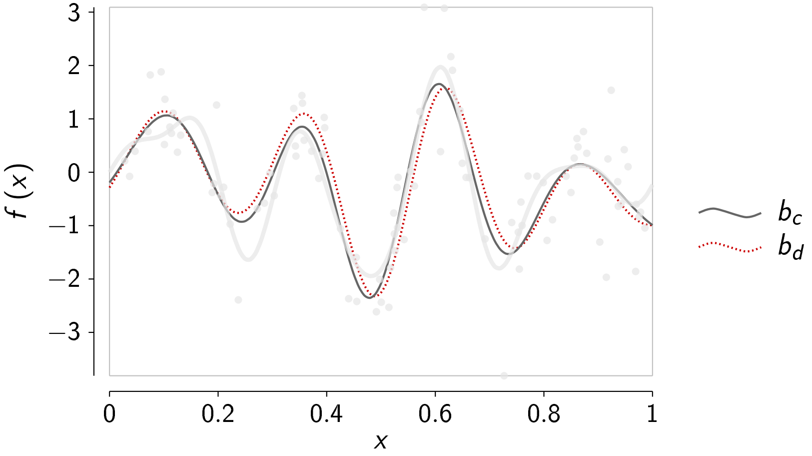
Regression strategy effectiveness example

$M = 100$, $E = 20$, $M_{\min} = 90$, $M_{\max} = 110$, $\text{SNR} \approx 2.5$



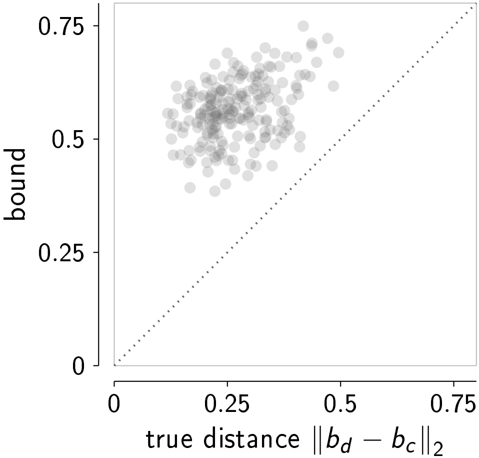
Regression strategy effectiveness example

$M = 100,\ E = 20,\ M_{\min} = 90,\ M_{\max} = 110,\ \mathrm{SNR} \approx 2.5$



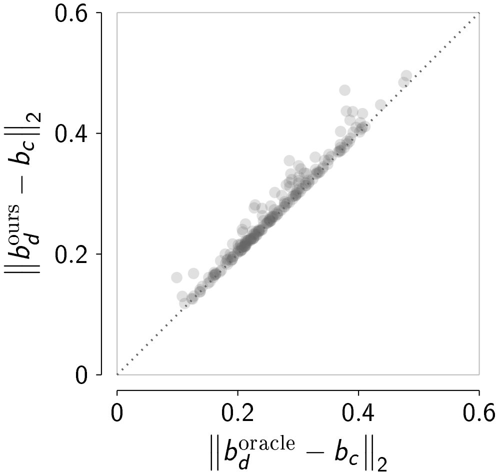
Accuracy of the computed bound

$M = 100, \ E = 20, \ M_{\min} = 90, \ M_{\max} = 110$



Comparison with oracle

$M = 100, \ E = 20, \ M_{\mathrm{min}} = 90, \ M_{\mathrm{max}} = 110$



Conclusions and future works for this part

Conclusions

Strategy:

- is effective and easy to be implemented
- has self-evaluation capabilities
- has self-tuning capabilities

Future works

- exploit statistical knowledge about M
- incorporate effects of finite number of steps in consensus algorithms
- extend to dynamic scenarios (long term objective)

Part Two

Privacy-aware number of agents estimation

Estimation of the number of agents (A) can be important in:

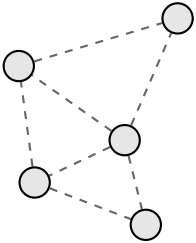
- distributed estimation
- analysis of connectivity

We assume *privacy concerns* $\rightarrow$ *do not use IDs!*

Our goal: obtain an easily implementable distributed estimator satisfying the constraints

The basic idea

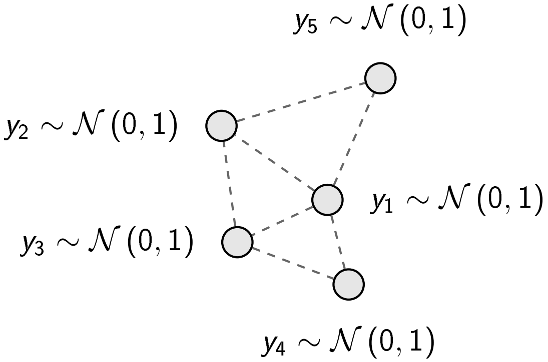
Algorithm:



The basic idea

Algorithm:

local generation



The basic idea

Algorithm:

local generation

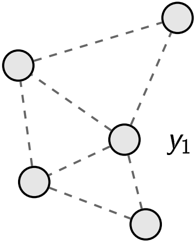
↓

average consensus

$$y_5 \to \frac{1}{A}\sum_{a=1}^{A} y_a$$

$$y_2 \to \frac{1}{A}\sum_{a=1}^{A} y_a$$

$$y_3 \to \frac{1}{A}\sum_{a=1}^{A} y_a$$



$$y_1 \to \frac{1}{A}\sum_{a=1}^{A} y_a$$

$$y_4 \to \frac{1}{A}\sum_{a=1}^{A} y_a$$

The basic idea

Algorithm:

local generation

↓

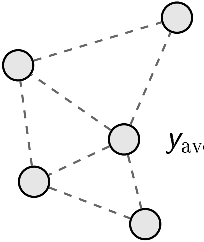
average consensus

↓

Maximum Likelihood → $\widehat{A^{-1}} = y_{\text{ave}}^2$

$y_{\text{ave}} \sim \mathcal{N}\left(0, \frac{1}{A}\right)$

$y_{\text{ave}} \sim \mathcal{N}\left(0, \frac{1}{A}\right)$



$y_{\text{ave}} \sim \mathcal{N}\left(0, \frac{1}{A}\right)$

$y_{\text{ave}} \sim \mathcal{N}\left(0, \frac{1}{A}\right)$

$y_{\text{ave}} \sim \mathcal{N}\left(0, \frac{1}{A}\right)$

The basic idea

Algorithm:

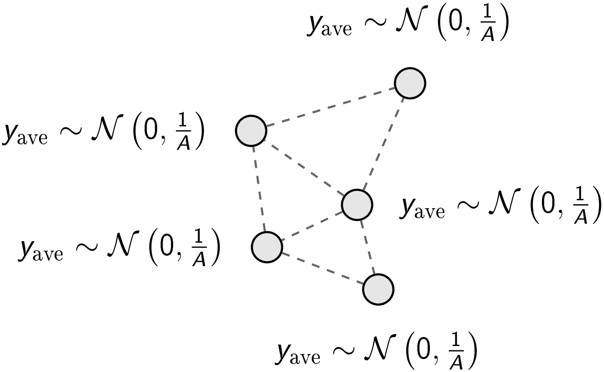
local generation

↓

average consensus

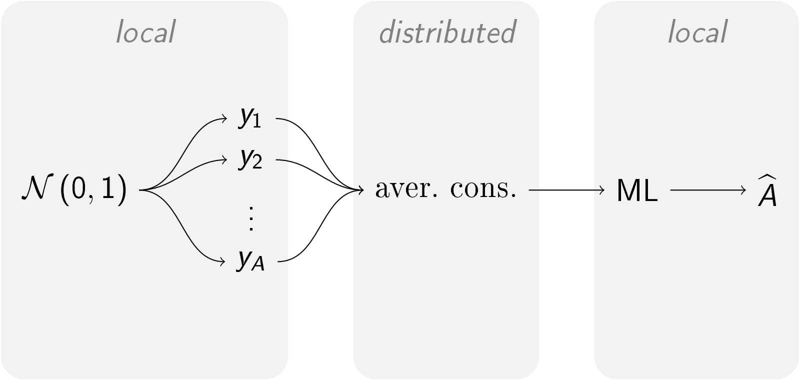
↓

Maximum Likelihood → $\widehat{A^{-1}} = y_{\text{ave}}^2$

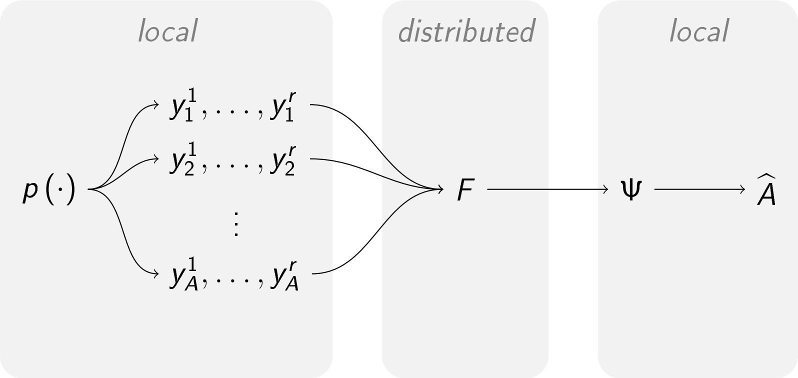


does not require sending IDs

Reformulating the idea as a block scheme



Plausible ways to generalize the idea



- $\mathcal{N}(\mu, \sigma^2)$
- $\mathcal{U}[\alpha, \beta]$
- ??

- average
- max
- ??

- ML
- MMSE
- MAP
- ??

Which cost function we consider

notice: we want to estimate A^{-1} instead of A

$$\downarrow$$

$$\text{estimator} =: \widehat{A^{-1}}$$

Considered cost function

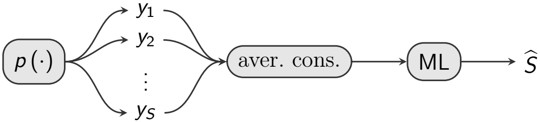
$$\mathbb{E}\left[\left(\widehat{A^{-1}} - A^{-1}\right)^2\right] \qquad \left(\equiv \text{variance if } \widehat{A^{-1}} \text{ unbiased}\right)$$

Why?

- convenient in order to obtain mathematical results
- in our cases, asymptotically in r:

$$\lim_{r \to +\infty} \mathbb{E}\left[\left(\frac{\widehat{A^{-1}} - A^{-1}}{A^{-1}}\right)^2\right] = \mathbb{E}\left[\left(\frac{\widehat{A} - A}{A}\right)^2\right]$$

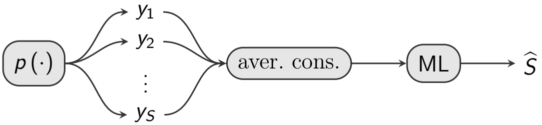
Theoretical results: average-consensus + ML



Assumptions

- $y_a^{\cdot}$ generated through Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$
- fusion of $y_a^{\cdot}$ is through average-consensus

Theoretical results: average-consensus + ML



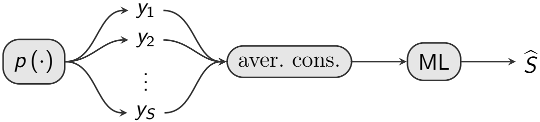
Assumptions

- $y_a^{\cdot}$ generated through Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$
- fusion of $y_a^{\cdot}$ is through average-consensus

Results: ML estimators:

- writable in closed form

Theoretical results: average-consensus + ML



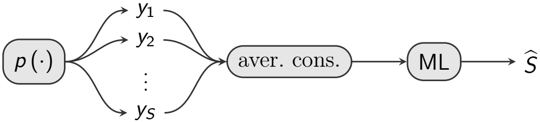
Assumptions

- $y_a^{\cdot}$ generated through Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$
- fusion of $y_a^{\cdot}$ is through average-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)

Theoretical results: average-consensus + ML



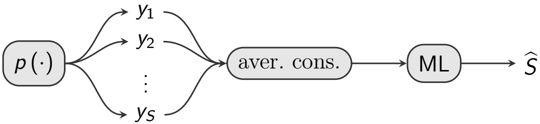
Assumptions

- $y_a^{\cdot}$ generated through Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$
- fusion of $y_a^{\cdot}$ is through average-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)
- performances: $\mathrm{var}\left(\dfrac{\widehat{A^{-1}} - A^{-1}}{A^{-1}}\right) = \dfrac{2}{r}$ (independent of μ and σ^2)

Theoretical results: average-consensus + ML



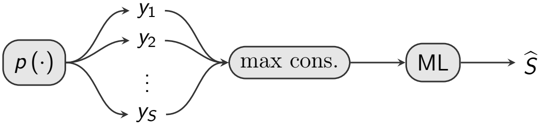
Assumptions

- $y_a^{\cdot}$ generated through Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$
- fusion of $y_a^{\cdot}$ is through average-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)
- performances: $\mathrm{var}\left(\dfrac{\widehat{A^{-1}} - A^{-1}}{A^{-1}}\right) = \dfrac{2}{r}$ (independent of μ and σ^2)
- (conjecture: Law of Large Numbers) if $r \to +\infty$ then performances are independent of $p(\cdot)$

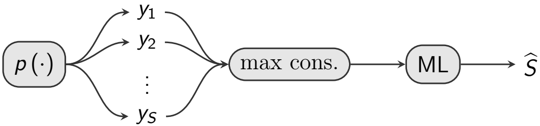
Theoretical results: max-consensus + ML



Assumptions

- cumulative distribution $P(\cdot)$ of $y_a^{\cdot}$ is strictly monotonic and continuous
- fusion of $y_a^{\cdot}$ is through max-consensus

Theoretical results: max-consensus + ML



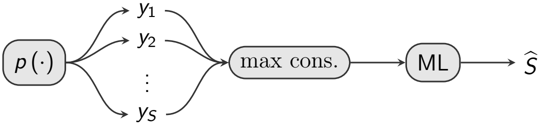
Assumptions

- cumulative distribution $P(\cdot)$ of $y_a^{\cdot}$ is strictly monotonic and continuous
- fusion of $y_a^{\cdot}$ is through max-consensus

Results: ML estimators:

- writable in closed form

Theoretical results: max-consensus + ML



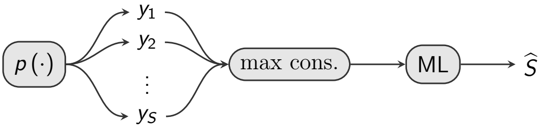
Assumptions

- cumulative distribution $P(\cdot)$ of $y_a^{\cdot}$ is strictly monotonic and continuous
- fusion of $y_a^{\cdot}$ is through max-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)

Theoretical results: max-consensus + ML



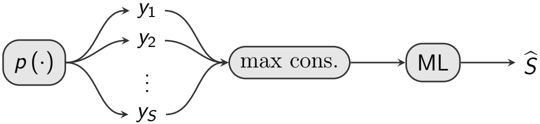
Assumptions

- cumulative distribution $P(\cdot)$ of $y_a^{\cdot}$ is strictly monotonic and continuous
- fusion of $y_a^{\cdot}$ is through max-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)
- performances: $\mathrm{var}\left(\frac{\widehat{A^{-1}} - A^{-1}}{A^{-1}}\right) = \frac{1}{r}$ independent of $P(\cdot)$

Theoretical results: max-consensus + ML



Assumptions

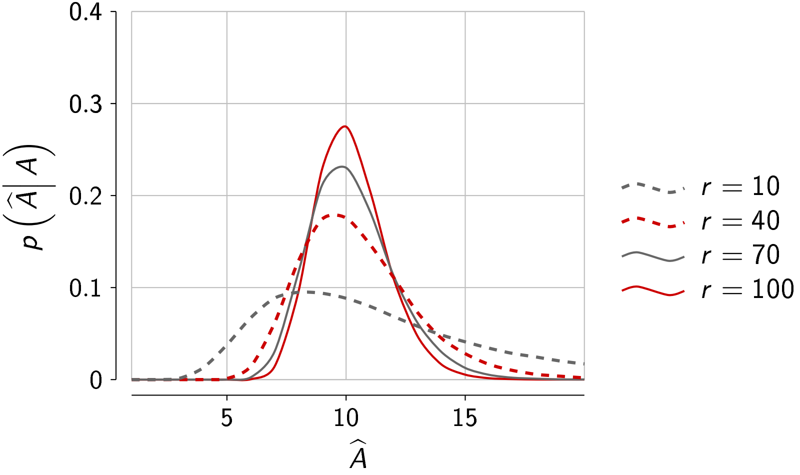
- cumulative distribution $P(\cdot)$ of y_a is strictly monotonic and continuous
- fusion of y_a is through max-consensus

Results: ML estimators:

- writable in closed form
- are MVUE (Minimum Variance and Unbiased)
- performances: $\mathrm{var}\left(\dfrac{\widehat{A^{-1}} - A^{-1}}{A^{-1}}\right) = \dfrac{1}{r}$ independent of $P(\cdot)$
- performances are twice as good as average-consensus

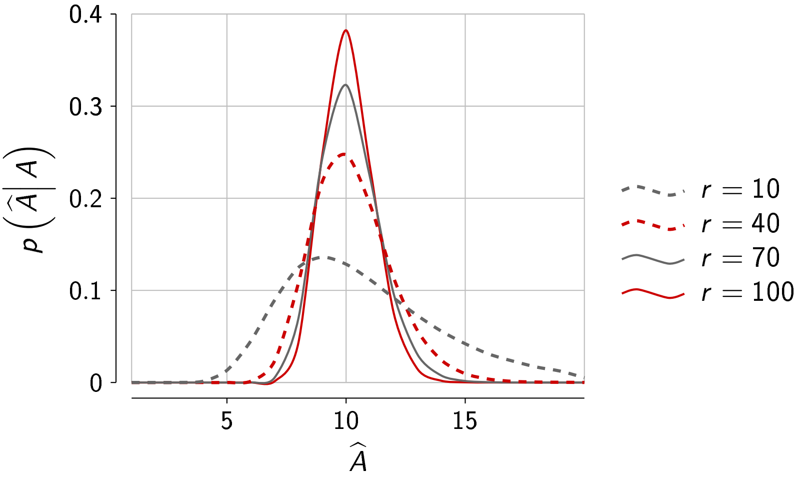
Results of various simulated systems (1)

$\mathcal{N}(0,1)$	average consensus	ML	A = 10



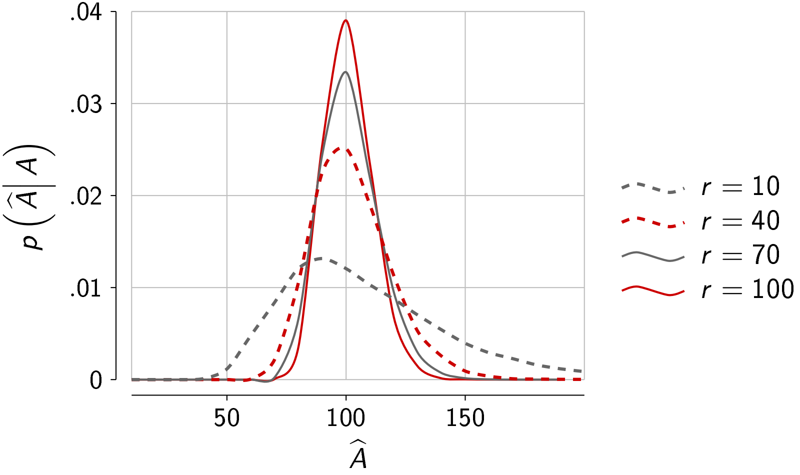
Results of various simulated systems (2)

$\mathcal{U}[0,1]$	max consensus	ML	$A = 10$



Results of various simulated systems (3)

$\mathcal{U}[0,1]$	max consensus	ML	A = 100



Conclusions

. . . and future extensions

Conclusions

- effective and robust algorithm
- quantifiable performances
- rely on statistical concepts $\rightarrow$ preserves privacy
- inherits good qualities of consensus strategies

Future extensions

- analyze optimal quantization strategies
- find optimal distributions for average consensus
- use the strategy for topological change detection purposes

Distributed Parametric-Nonparametric Estimation in Networked Control Systems

Damiano Varagnolo

Department of Information Engineering - University of Padova

April, 18^{th} 2011

damiano.varagnolo@dei.unipd.it
www.dei.unipd.it/∼varagnolo/
google: damiano varagnolo

entirely written in LaTeX2ε using Beamer and TikZ

Distributed Optimization

- F. Zanella, D. Varagnolo, A. Cenedese, G. Pillonetto, L. Schenato. Newton-Raphson consensus for distributed convex optimization. IEEE Conference on Decision and Control (CDC 2011) (submitted)

Applications of Consensus

- S. Bolognani, S. Del Favero, L. Schenato, D. Varagnolo. Consensus-based distributed sensor calibration and least-square parameter identification in WSNs. International Journal of Robust and Nonlinear Control, 2010
- S. Bolognani, S. Del Favero, L. Schenato, D. Varagnolo. Distributed sensor calibration and least-square parameter identification in WSNs using consensus algorithms. Proceedings of Allerton Conference on Communication Control and Computing (Allerton 2008)

Parametric Regression

- D. Varagnolo, G. Pillonetto, L. Schenato. Distributed consensus-based Bayesian estimation: sufficient conditions for performance characterization. 2010 American Control Conference, 2010
- D. Varagnolo, P. Chen, L. Schenato, S. Sastry. Performance analysis of different routing protocols in wireless sensor networks for real-time estimation. IEEE Conference on Decision and Control (CDC 2008)

Nonparametric Regression / Classification

- D. Varagnolo, G. Pillonetto, L. Schenato. Distributed parametric and nonparametric regression with on-line performance bounds computation. Automatica (submitted)
- D. Varagnolo, G. Pillonetto, L. Schenato. Auto-tuning procedures for distributed nonparametric regression algorithms. IEEE Conference on Decision and Control (CDC 2011) (submitted)
- S. Del Favero, D. Varagnolo, F. Dinuzzo, L. Schenato, G. Pillonetto. On the discardability of data in Support Vector Classification problems. IEEE Conference on Decision and Control (CDC 2011) (submitted)
- D. Varagnolo, G. Pillonetto, L. Schenato. Distributed Function and Time Delay Estimation using Nonparametric Techniques. IEEE Conference on Decision and Control (CDC 2009)

Number of Sensors Estimation

- D. Varagnolo, G. Pillonetto, L. Schenato. Distributed statistical estimation of the number of nodes in Sensor Networks. IEEE Conference on Decision and Control (CDC 2010)

Smart Grids

- E. Bitar, A. Giani, R. Rajagopal, D. Varagnolo, P. Khargonekar, K. Poolla, V. Pravin. Optimal Contracts for Wind Power Producers in Electricity Markets. Conference on Decision and Control CDC10, 2010

left mouse / page down / space	next slide
right mouse / page up / backspace	previous slide
q / esc	exit
tab	overview toggling
t	clock toggling
home	go to first slide
end	go to last slide
l	go to the previously last seen slide
b	fade the screen to black
w	fade the screen to white
f	full screen toggling
enter	spotlight toggling
+ / -	adjust the spotlight size
mouse wheel	adjust the spotlight size
left mouse (dragging a box)	highglight a box
right mouse (on a highlighted box)	remove the highlight of that box
z	zoom toggling
right mouse (in zoom modality)	move on the image