Smart Multi-agent Control Systems over wireless: Challenges and Perspectives

Luca Schenato
University of Padova

M.Ag.I.C.
Multi Agent Intelligent Control

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Outline

- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

Joint work with

Research fellows

Ruggero Carli

Gianluigi Pillonetto

Subhrakanti Dey
(Univ South Australia)

Current Ph.D/post-docs:

Marco Todescato
(soon at Bosch, Germany)

Nicoletta Bof

Enrica Rossi

Former Ph.D/post-docs:

Damiano Varagnolo
Lulea Univ., Sweden

Guido Cavraro
Virginia Tech, USA

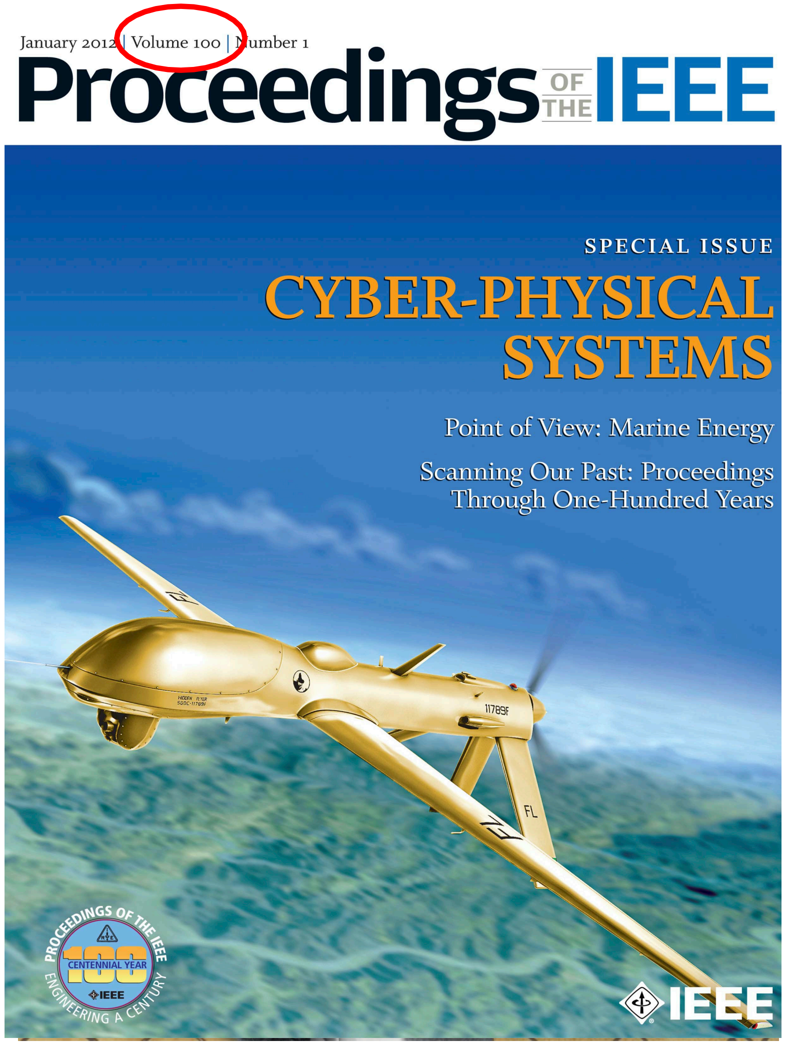
Andrea Carron
ETH, Switzerland

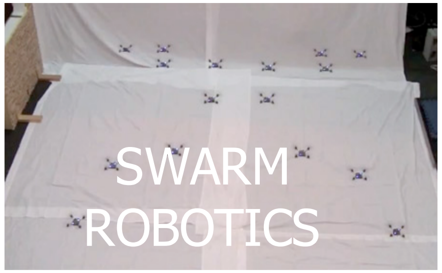
DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

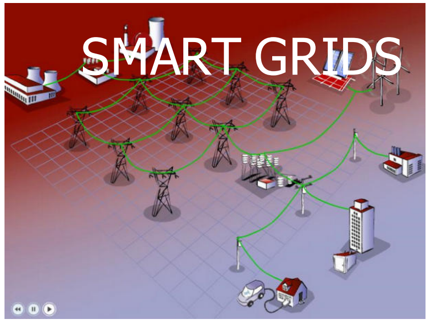
Outline

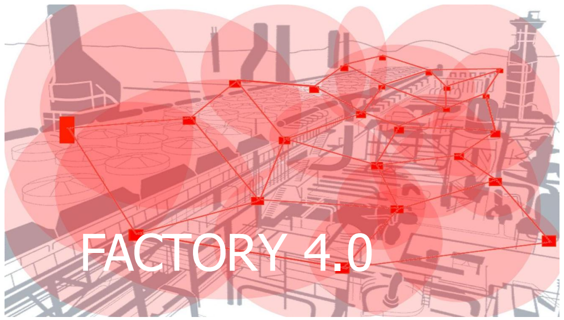
- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

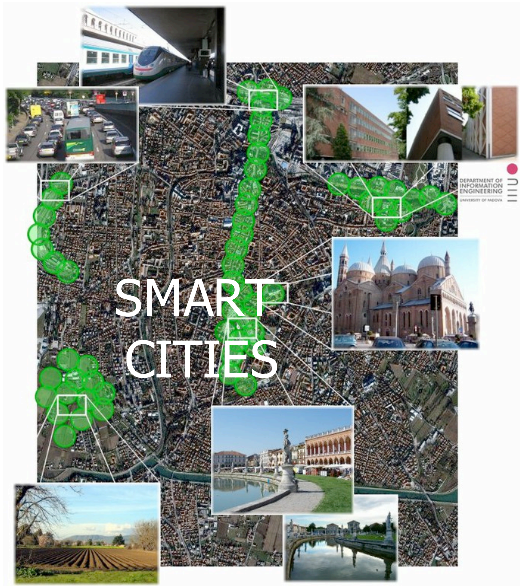
The XXI century: a Smart World



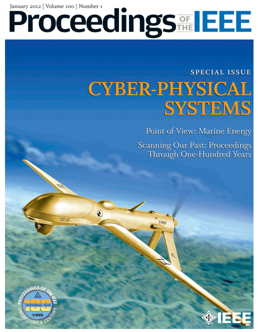


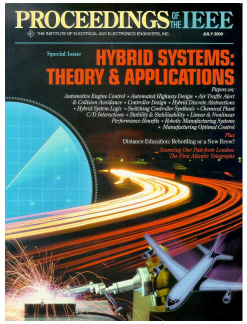


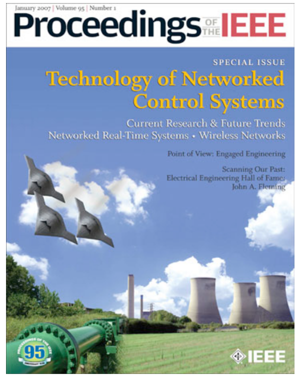


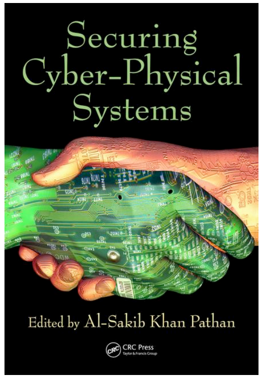


The ICT scientific army

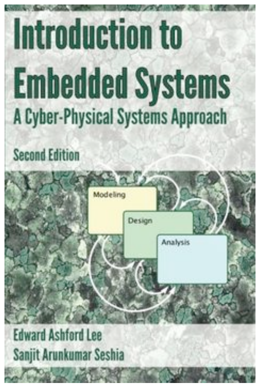


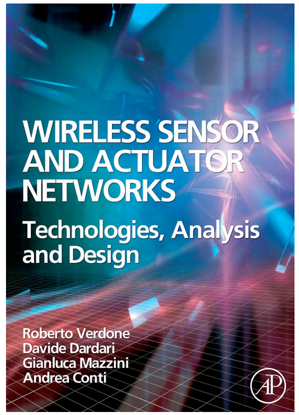


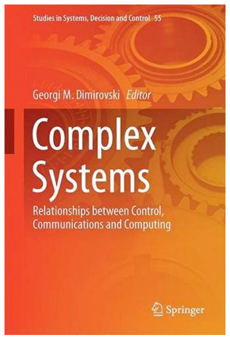


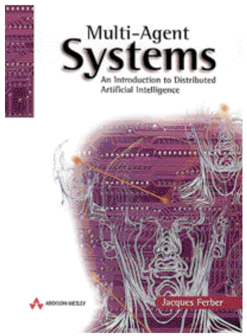


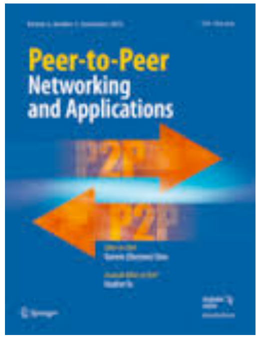
Before 2000's	Today		Tomorrow
			
Centralized Reliable comm.	Multi-agent Reliable comm.	Centralized Unrelia. Comm.	Multi-agent Unrelia. Comm.
✓	✓	✓	?

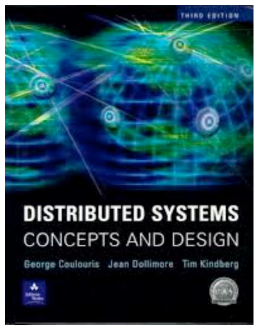






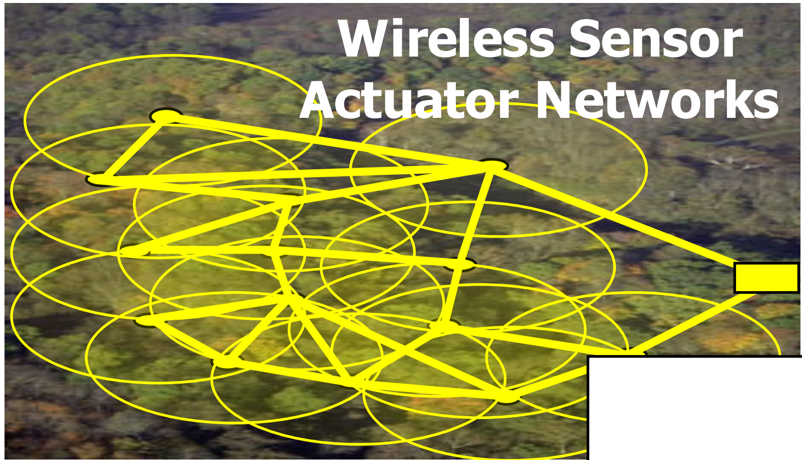


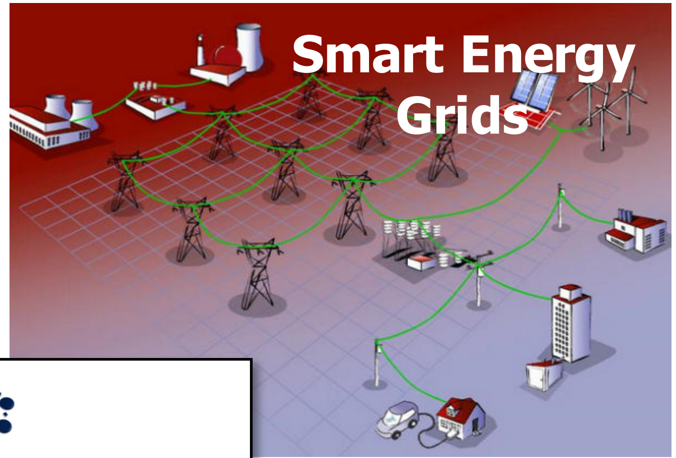




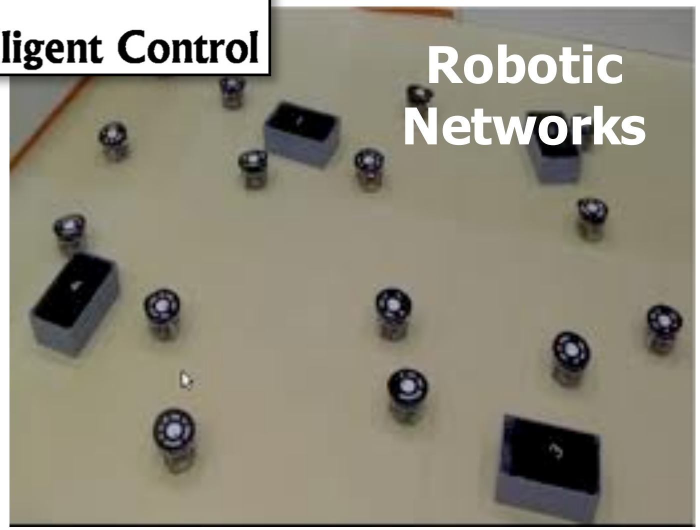
DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Target applications: MAgIC Lab. at University of Padova

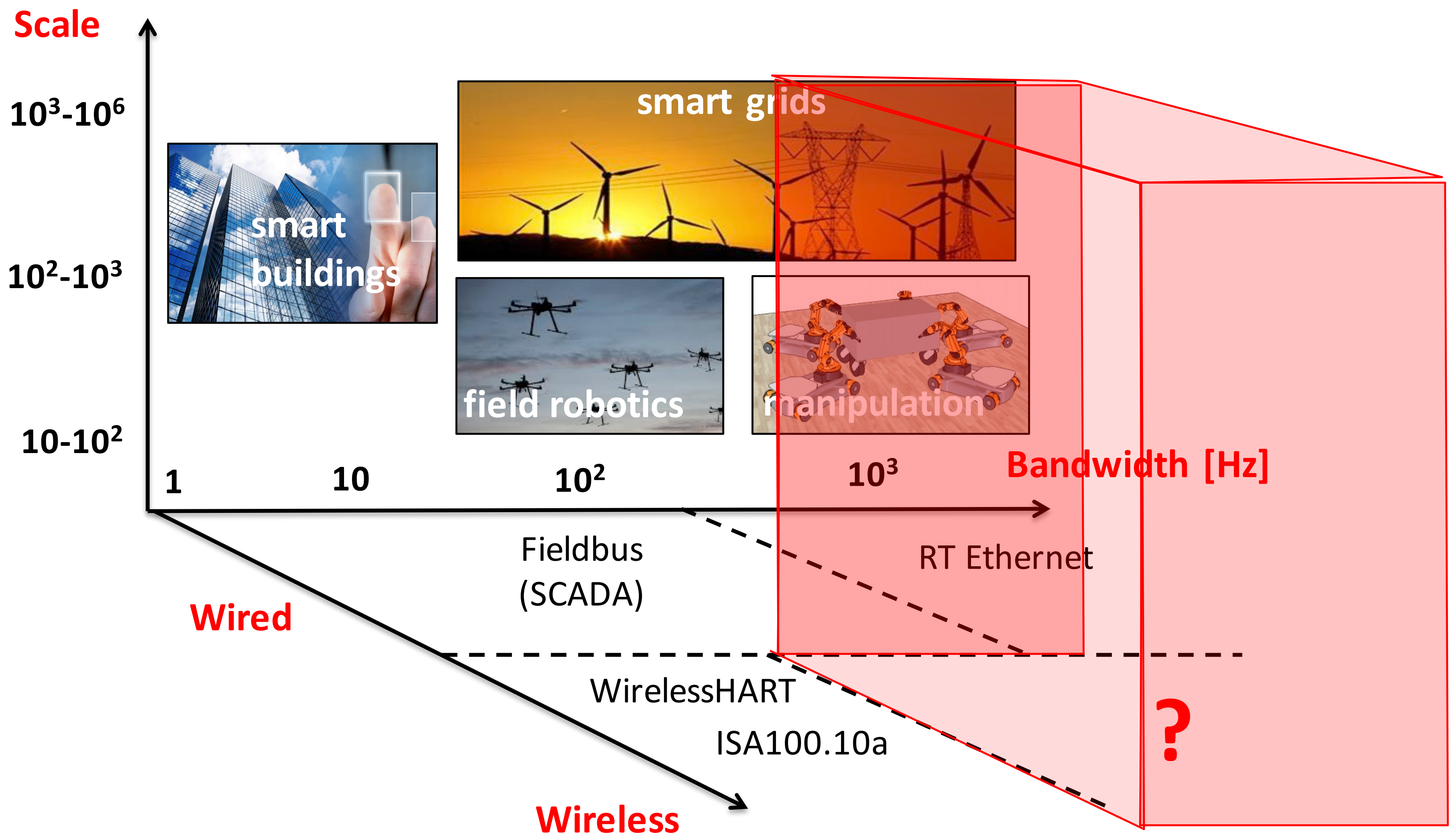




M.A.g.I.C.
Multi Agent Intelligent Control



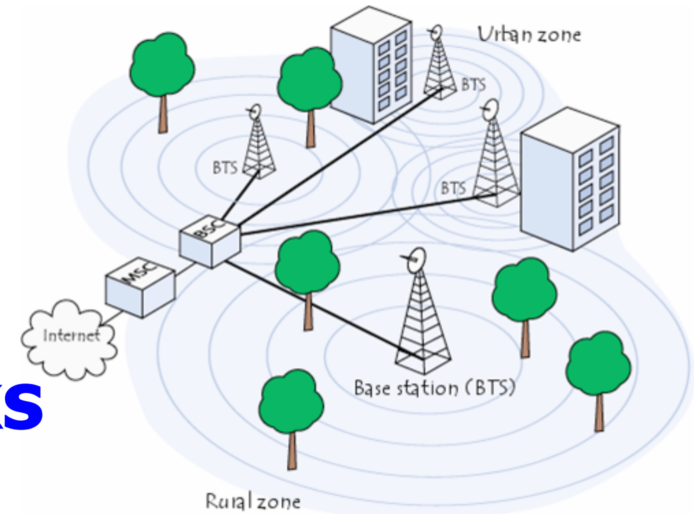
The challenge cube for time-critical smart systems



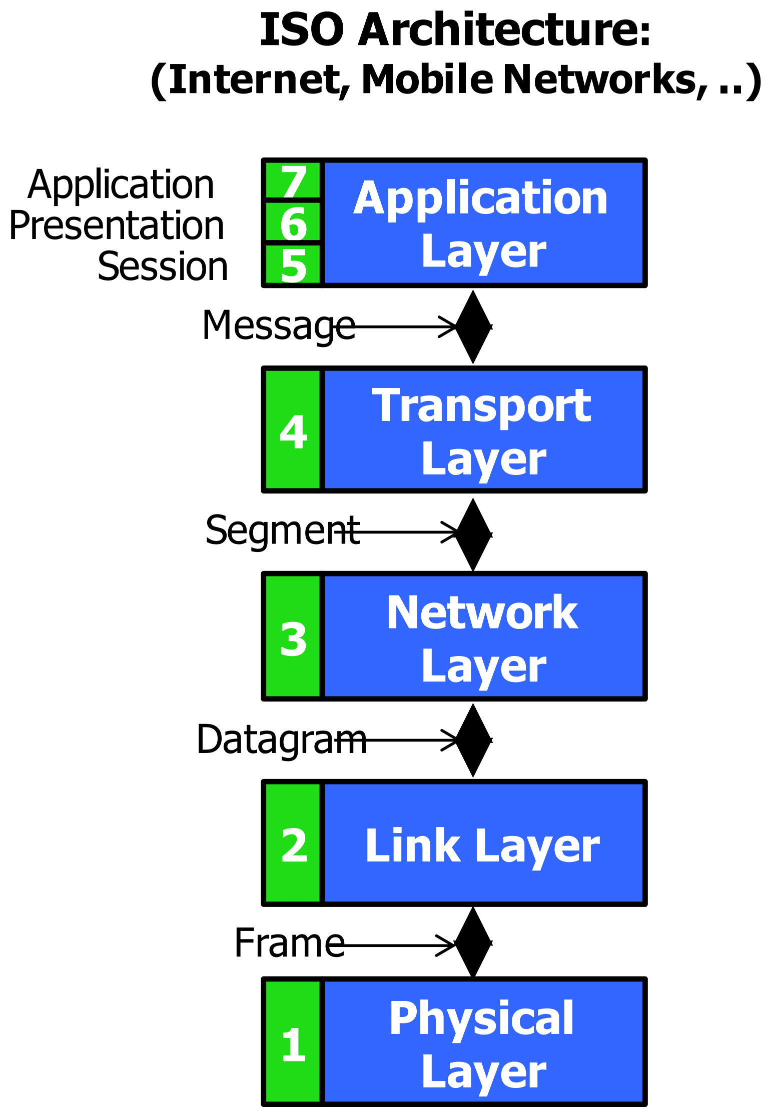
Some working complex systems

INTERNET

Cell phones networks



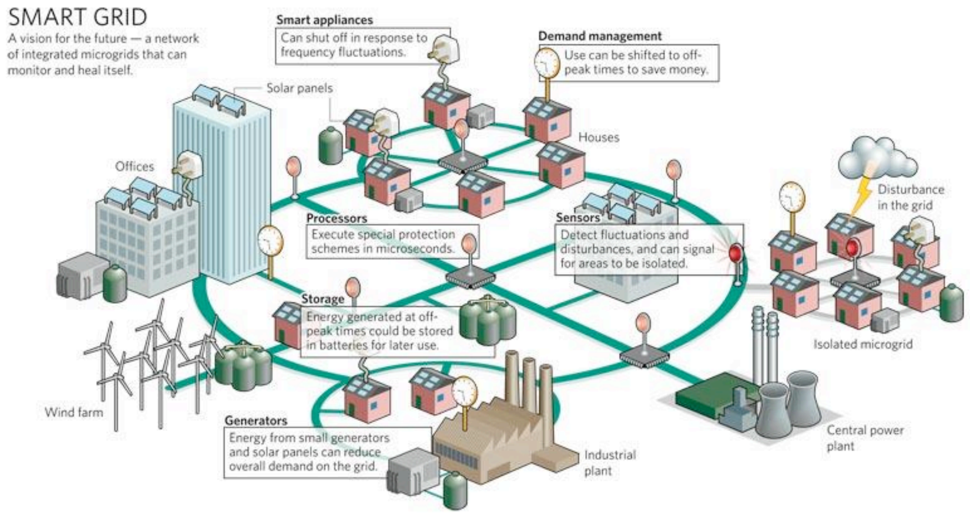
A leading paradigm: ISO layers with few primitives



DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Smart multi-agent systems: an ISO-like paradigm ?

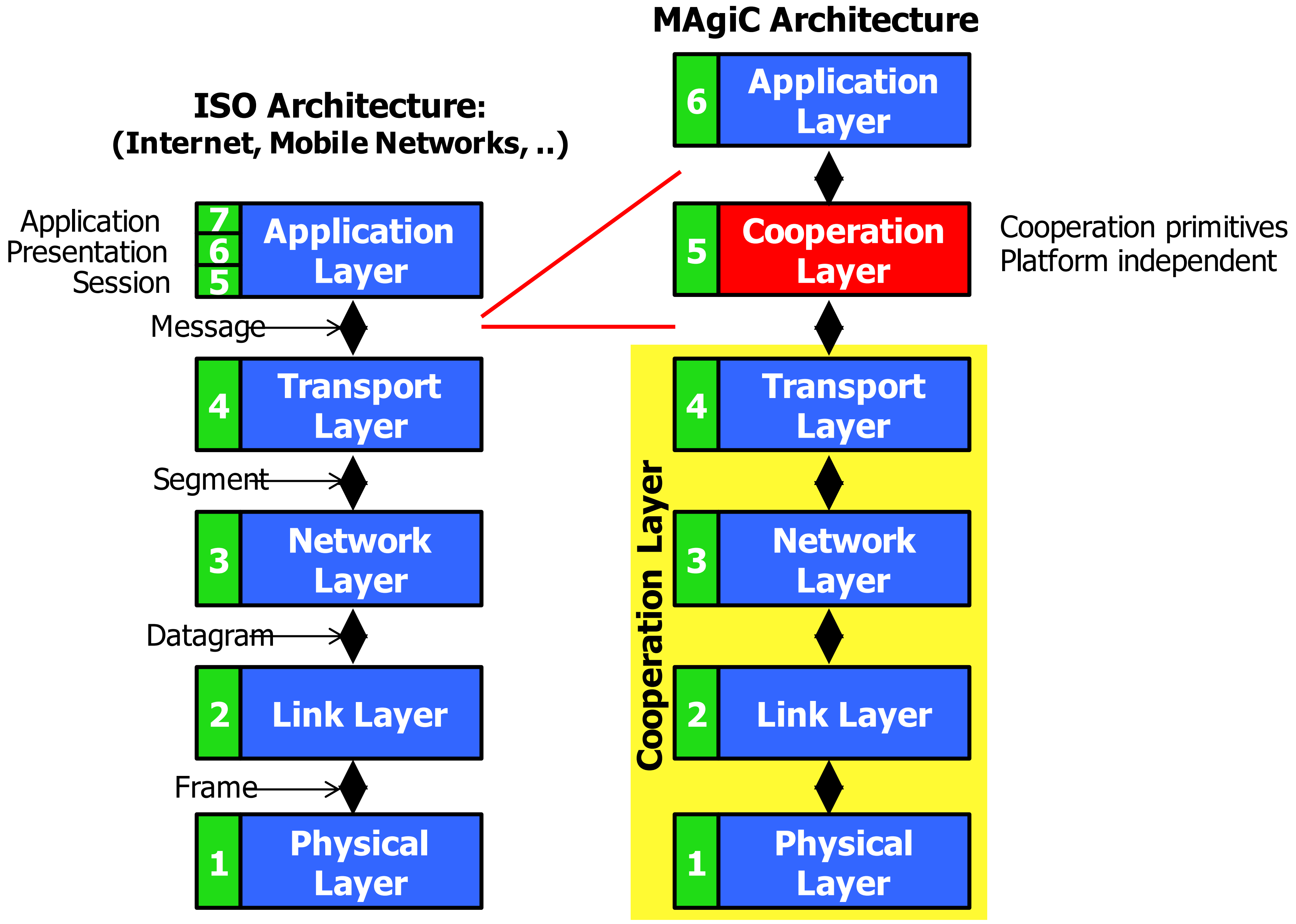
Smart Power Grids



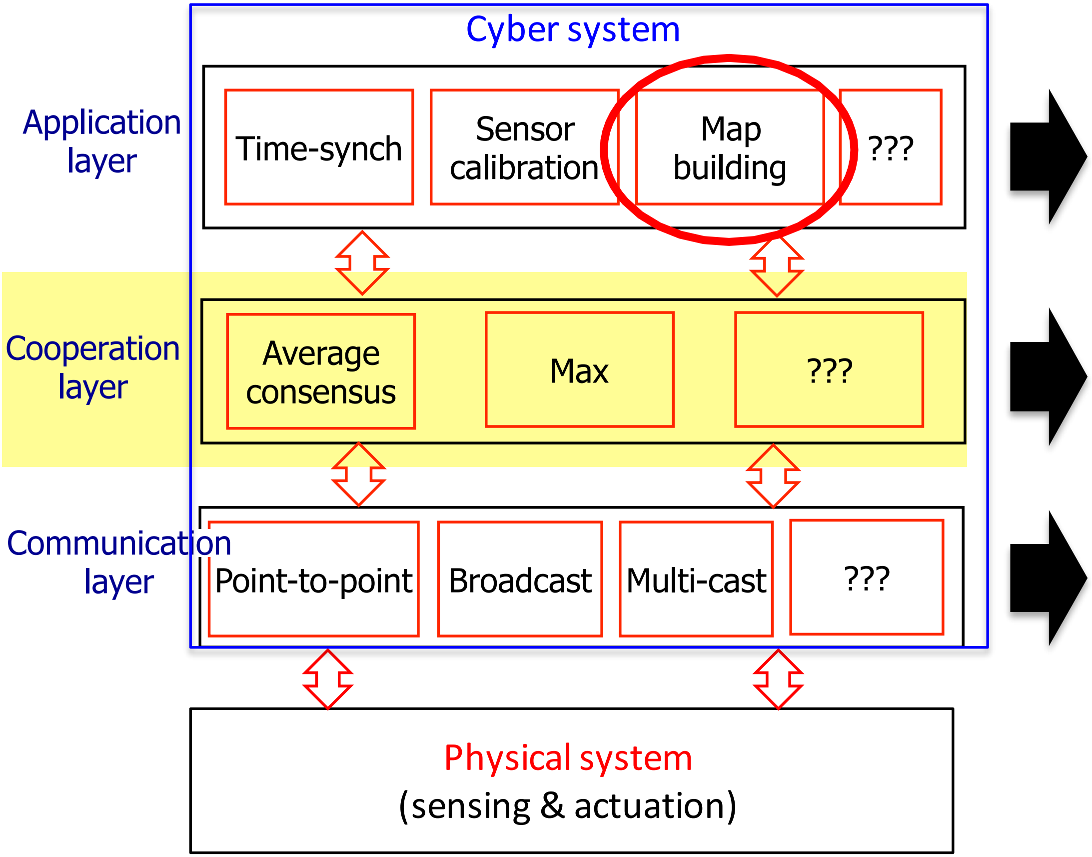
Intelligent transportation

- Need to seamlessly integrate:
  - Communication network(s)
  - Sensing and control
  - Physical constraints (conservation mass/energy, dynamics)
  - Social constraints (markets, policies)

A leading paradigm: ISO layers with few primitives



Current research agenda



- **Machine-learning** estimation
- Casting cooperative estimation & control as optimization problems

- **Distributed optimization** over unreliable communication

- WiFi-based protocols for control
- Adaptive **communication&control** rate via cross-layer desing

Interdisciplinary Approach

Outline

- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

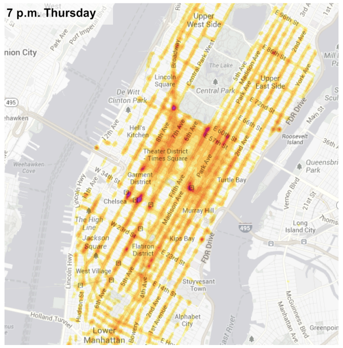
Learning problems:

Density estimation:

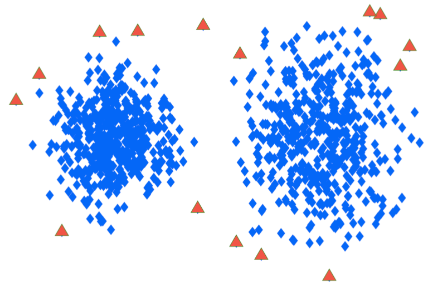
$f(x) : \mathbb{R}^n \to \mathbb{R}$

$f(x) \geq 0, \quad \int f(x) = 1$

$\mathcal{D} = \{x_1, x_2, ...\}$: events



Taxi pick-up calls

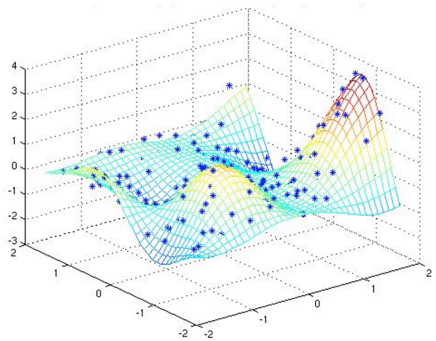

Anomaly detection

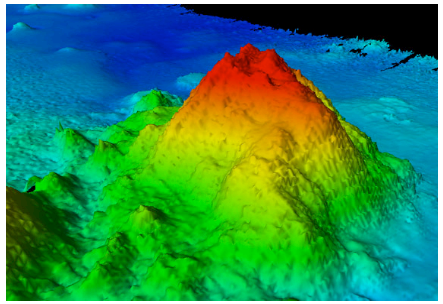
Regression:

$f(x) : \mathbb{R}^n \to \mathbb{R}$

$y_i = f(x_i) + v_i$

v_i noise


Pollution level profile

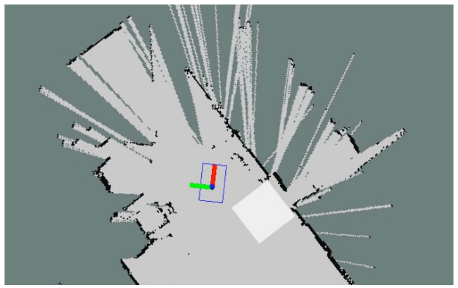

Seabed depth profile

Classification

$f(x) : \mathbb{R}^n \to \{0, 1\}$

$y_i = f(x_i) + v_i$

v_i noise


Obstacles map


Oil-spill boundary

Multi-agent regression

- Parametric vs non-parametric

$f(x) = \sum_{i=1}^{m} \theta_i g_i(x)$

$\theta \in \mathbb{R}^m$, unknown

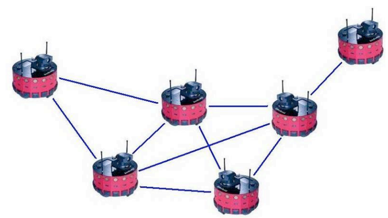
$g_i(x)$ known

$f(x) \in RKHS$, infinite dimensional

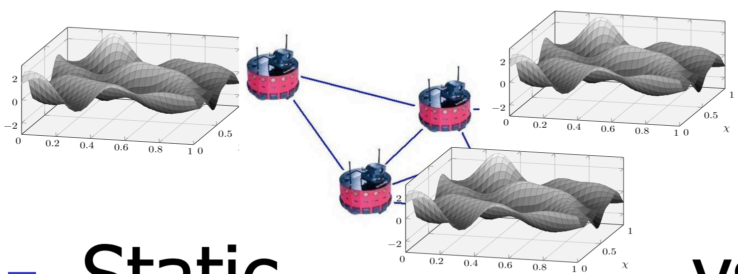
$f(x)$ defined via Kernel $k(x, x')$

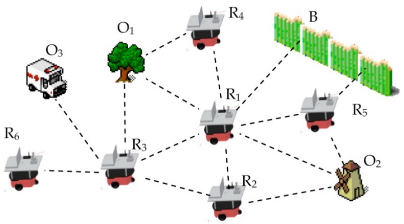
$k(x, x')$ known

- Cloud-based vs peer-to-peer

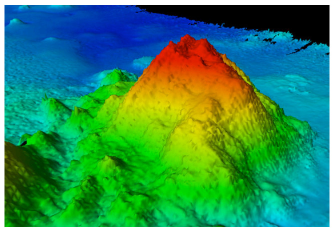


- Global vs Local estimation

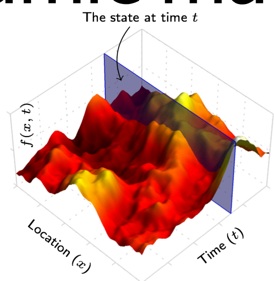




- Static vs dynamic maps:



$f(x)$



$f(x, t)$

Multi-agent regression

- Parametric vs non-parametric

$f(x) = \sum_{i=1}^{m} \theta_i g_i(x)$

$\theta \in \mathbb{R}^m$, unknown

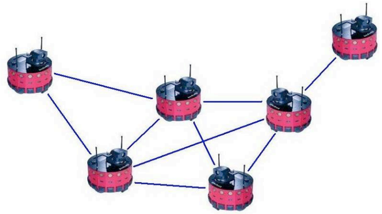
$g_i(x)$ known

$f(x) \in RKHS$, infinite dimensional

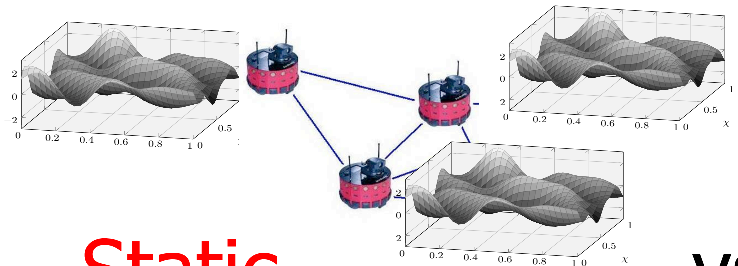
$f(x)$ defined via Kernel $k(x, x')$

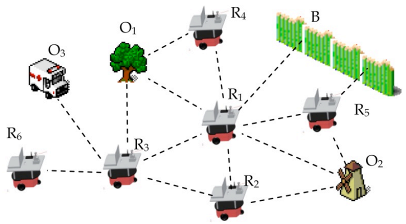
$k(x, x')$ known

- Cloud-based vs peer-to-peer

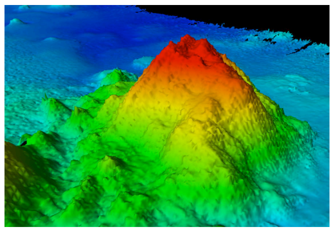


- Global vs Local estimation

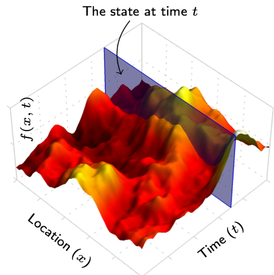




- Static vs dynamic maps:



$f(x)$

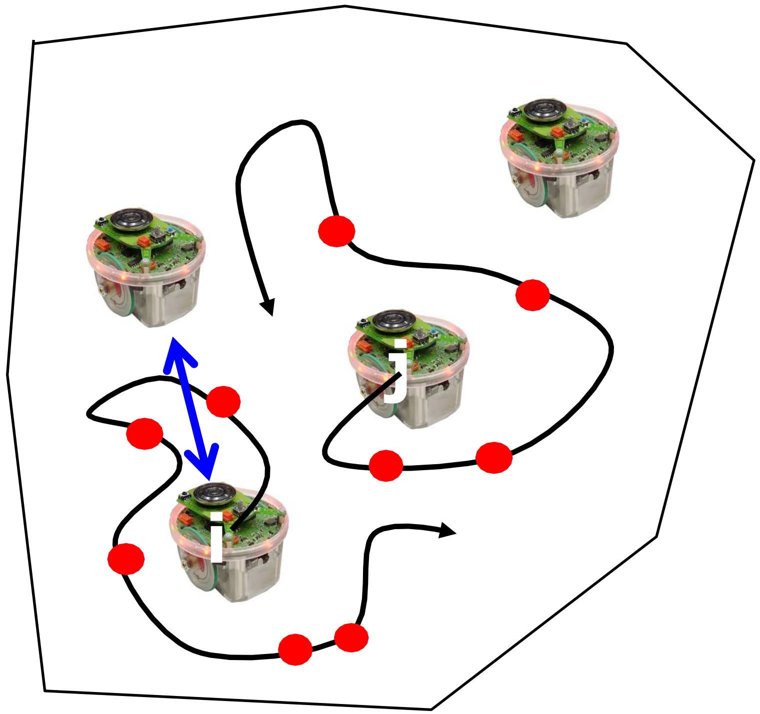


$f(x, t)$

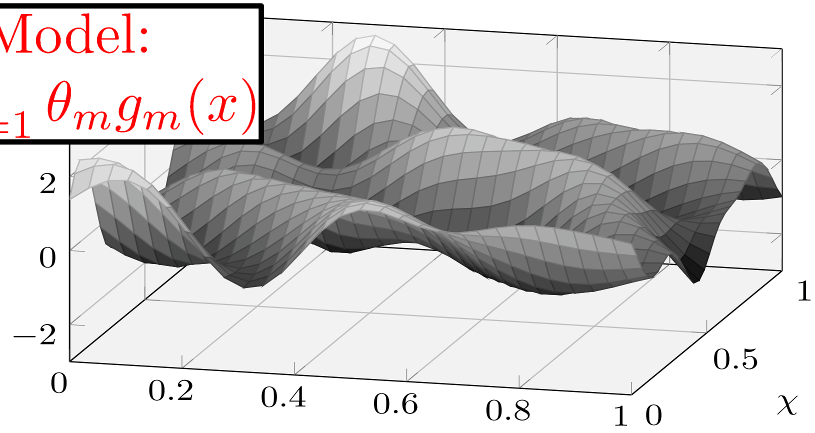
Outline

- Motivations, target applications & challenges
- Parametric regression
- Non-parametric regression
- Semi non-parametric regression
- Non-parametric regression for dynamical systems
- Conclusion and open problems

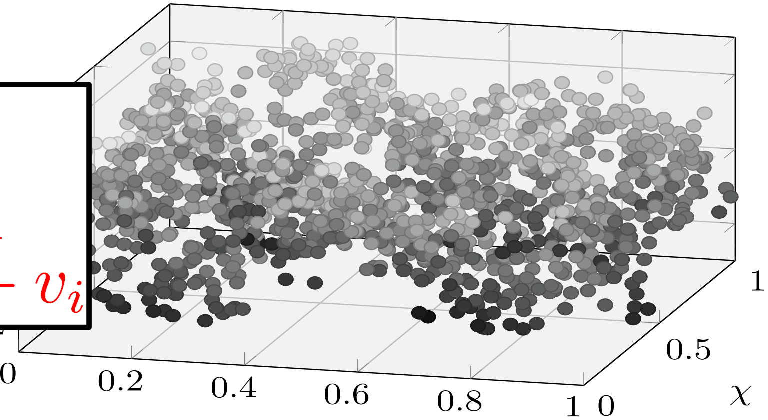
Example: Map-building in robotic networks



Parametric Model:
$f(x) = \sum_{m=1}^{M} \theta_m g_m(x)$



Noisy data:
$\{(x_i, y_i)\}_{i=1}^{N}$
$y_i = f(x_i) + v_i$



Goal:
$\min_\theta \sum_i v_i^2$

Map-building as least-squares regression

- Model class:

$$f(x) = \sum_{m=1}^{M} \theta_m g_m(x)$$

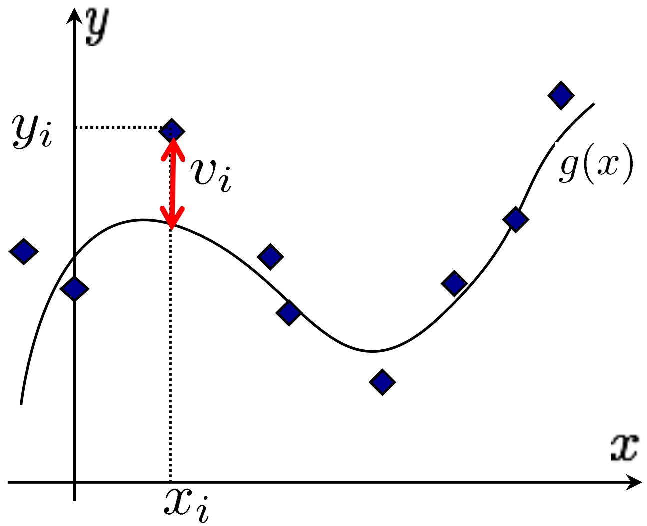
- Noisy measurements:

$$y_i = \sum_{m=1}^{M} \theta_m g_m(x_i) + v_i, \quad i = 1, \dots, N$$

$$\begin{bmatrix} y(x_1) \\ y(x_2) \\ \vdots \end{bmatrix} = \begin{bmatrix} g_1(x_1) & \dots & g_M(x_1) \\ \vdots & \vdots & \vdots \\ g_1(x_N) & \dots & g_M(x_N) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_M \end{bmatrix} + \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix}$$

G_i^T

$$y = G\theta + v$$



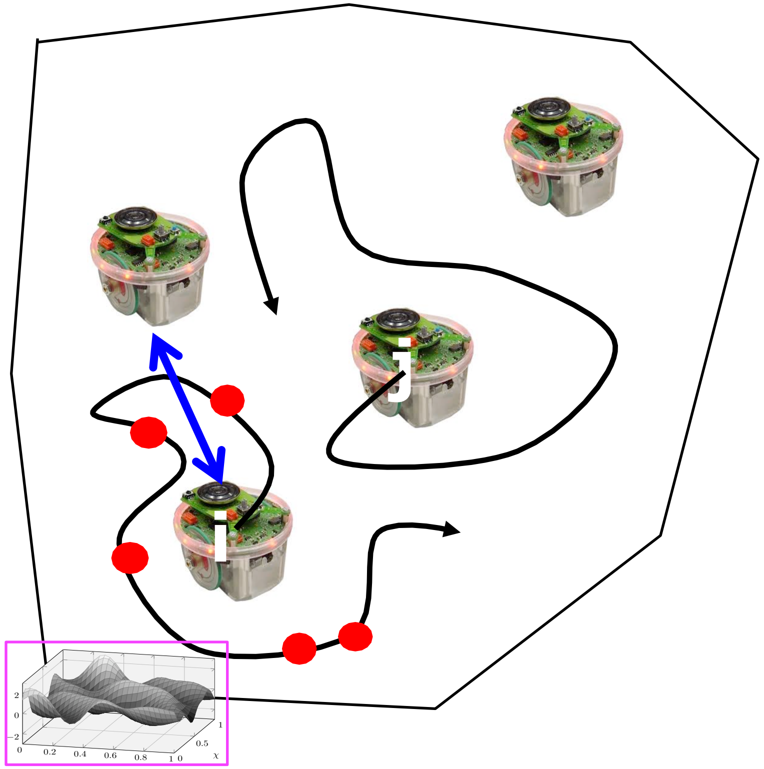
- Goal: minimize sum of squares of residues

$$\hat{\theta} = \text{argmin}_\theta \sum_{i=1}^{N} v_i^2$$

$$\hat{\theta} = \left(\sum_{i=1}^{N} G_i G_i^T\right)^{-1}\left(\sum_{i=1}^{N} G_i y_i\right) = \left(\frac{1}{N}\sum_{i=1}^{N} G_i G_i^T\right)^{-1}\left(\frac{1}{N}\sum_{i=1}^{N} G_i y_i\right)$$

- Xiao-Boyd-Lall, 2005
- Bolognani-Del Favero-Schenato-Varagnolo, 2010

Consensus-based Map-building: gossip communication



ALGORITHM:

1) Initialize statistics:

$$Z_0^i = 0 \in R^{M\times M}$$
$$z_0^i = 0 \in R^{M}$$

2) Collect data and build local statistics:

$$Z_{t+1}^i = Z_i^t + G_t^i {G_t^i}^T$$
$$z_{t+1}^i = z_i^t + G_t^i y_t^i$$

3) Choose neighbor j and do gossip consensus:

$$Z_{t+1}^j = Z_{t+1}^i = \frac{1}{2} Z_t^i + \frac{1}{2} Z_t^j$$
$$z_{t+1}^j = z_{t+1}^i = \frac{1}{2} z_i^t + \frac{1}{2} z_j^t$$

4) Estimate map:

$$\hat{\theta}_t^i = (Z_t^i)^{-1} z_t^i$$

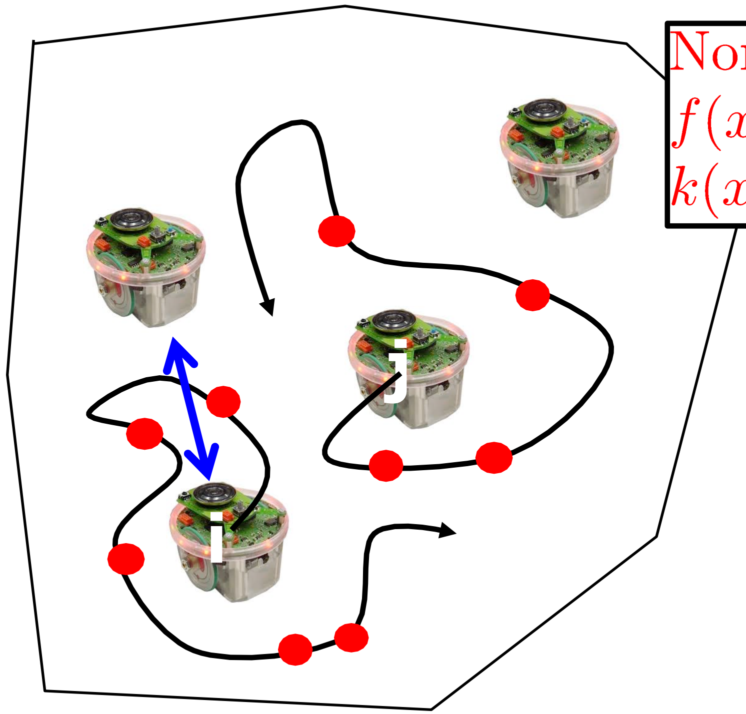
5) Repeat steps 2,3,4 (non necessarely in order)

- PROS:
  - Can be distributed
  - Gradient-based implementation: ADMM, gradient-consensus,
  - Extension to robust costs, e.g. $|| \; ||_1$
- CONS:
  - How to select basis functions
  - No estimate unless at least M data
  - Gradient-based implementations require step-size design

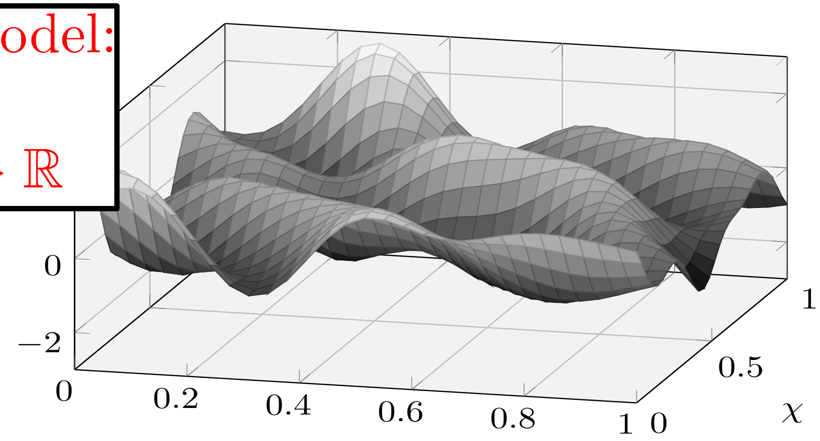
Outline

- Motivations, target applications & challenges
- Parametric regression
- Non-parametric regression
- Semi non-parametric regression
- Non-parametric regression for dynamical systems
- Conclusion and open problems

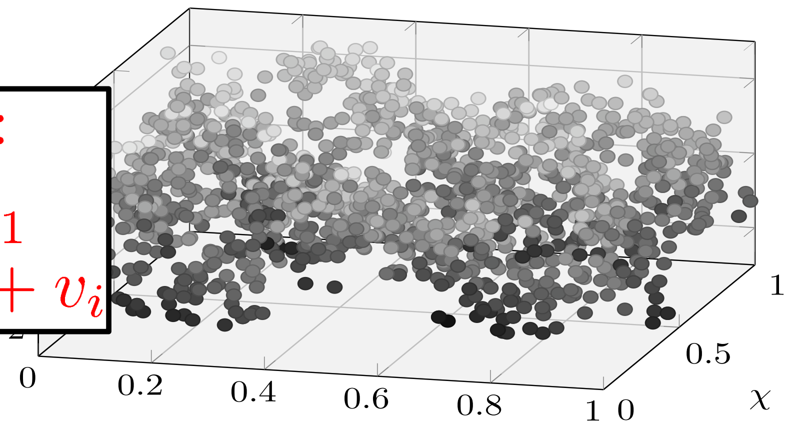
Gaussian regression (non parametric)



Non-parametric Model:
$f(x) \in \mathrm{RKHS}$
$k(x, x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$



Noisy data:
$\{(x_i, y_i)\}_{i=1}^{N}$
$y_i = f(x_i) + v_i$



Goal:
$\min_\theta \sum_i v_i^2$

Reproducing Kernel Hilbert Spaces (RKHS) (con't)

$k(x, x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$: Mercel Kernel
1) $k(\cdot, \cdot)$ continous, $\mathcal{X}$: compact
2) symmetric: $k(x, x') = k(x', x)$
3) positive semidefinite: $K \in \mathbb{R}^{N\times N} \geq 0, \quad [K]_{i,j} = k(x_i, x_j), \forall x_i, \forall N,$

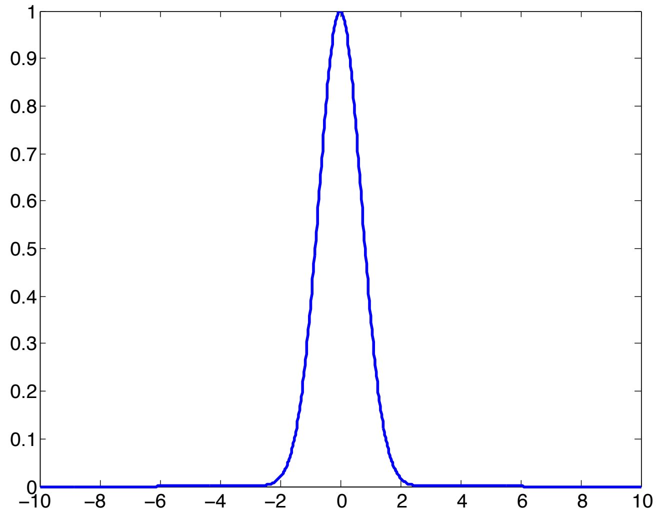
Bayesian Interpretation:

$\mathbb{E}[f(x)] = 0, \quad \mathbb{E}[f(x)f(x')] = k(x, x')$: zero-mean gaussian process

correlation

$\{(x_i, y_i)\}_{i=1}^N$: Noisy data:
$y_i = f(x_i) + v_i, \quad v_i \sim \mathcal{N}(0, \sigma^2)$

$$k(x, x') = \lambda e^{-\frac{\|x-x'\|^2}{2\sigma^2}}$$



Reproducing Kernel Hilbert Spaces (RKHS) (con't)

$k(x,x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$: Mercel Kernel
1) $k(\cdot,\cdot)$ continous, $\mathcal{X}$: compact
2) symmetric: $k(x,x') = k(x',x)$
3) positive semidefinite: $K \in \mathbb{R}^{N \times N} \geq 0, \quad [K]_{i,j} = k(x_i, x_j), \forall x_i, \forall N,$

Bayesian Interpretation:

$\mathbb{E}[f(x)] = 0, \quad \mathbb{E}[f(x)f(x')] = k(x,x')$: zero-mean gaussian process

$\{(x_i, y_i)\}_{i=1}^N$: Noisy data:
$y_i = f(x_i) + v_i, \quad v_i \sim \mathcal{N}(0, \sigma^2)$

$$\hat{f}(x) = \mathbb{E}[f(x) \,|\, \{x_i, y_i\}_{i=1}^M] = \sum_{i=1}^N c_i k(x_i, x)$$

$$\begin{bmatrix} c_1 \\ \vdots \\ c_M \end{bmatrix} = \left(K + \sigma^2 I\right)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}, \quad K := \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_M) \\ \vdots & & \vdots \\ k(x_M, x_1) & \cdots & k(x_M, x_M) \end{bmatrix}$$

Parametric vs non-parametric

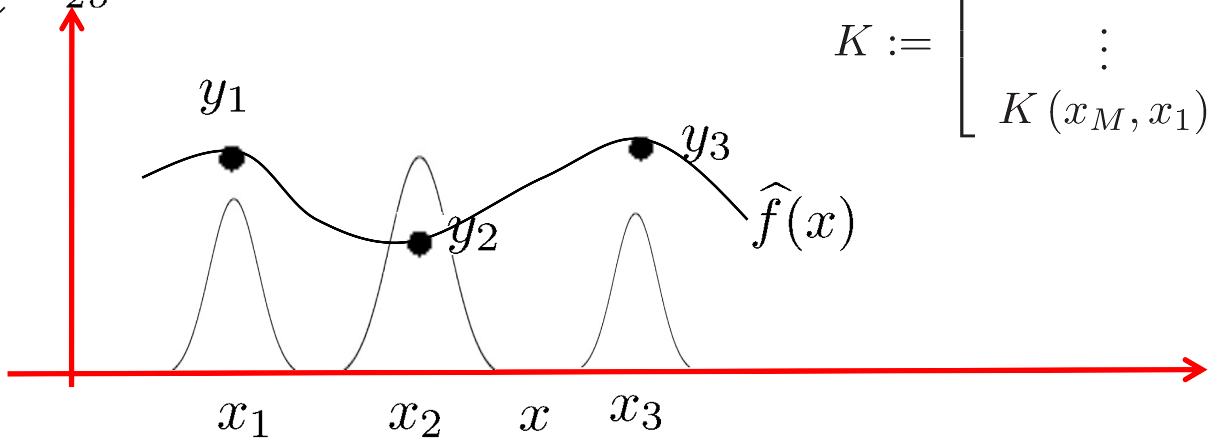
$$\{(x_i, y_i)\}_{i=1}^N: \text{ Noisy data:}$$
$$y_i = f(x_i) + v_i, \quad v_i \sim \mathcal{N}(0, \sigma^2)$$

$$f(x) = \sum_{i=1}^M \theta_i g_i(x)$$

$$k(x, x') = e^{\frac{|x-x'|^2}{2\sigma^2}}$$

$$g_i(x) = e^{\frac{|x-x_i|^2}{2\sigma^2}}$$

$$K := \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_M) \\ \vdots & & \vdots \\ K(x_M, x_1) & \cdots & K(x_M, x_M) \end{bmatrix}$$



$$\widehat{f}(x) = \sum_{i=1}^M \widehat{\theta}_i g_i(x)$$

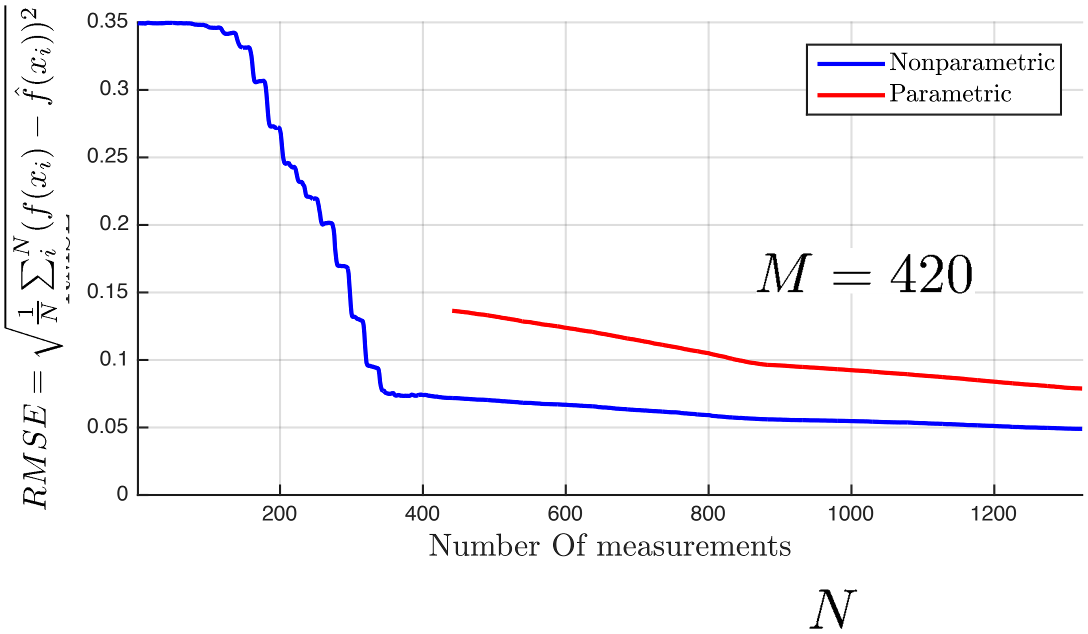
$$\widehat{f}(x) = \sum_{i=1}^M \widehat{c}_i k(x_i, x) = \sum_{i=1}^M \hat{c}_i g_i(x)$$

$$\widehat{\theta} = K^{-1} y$$

$$\widehat{c} = (K + \sigma^2 I)^{-1} y$$

regularization term

Parametric vs non-parametric



	PARAMETRIC	NON-PARAMETRIC
PROS	• Distributed (consensus) • Bounded complexity $O(M^3)$	• Better performance • Adaptable resolution
CONS	• What $g_i(x)$? • Need N>M points • Over-fitting & ill-conditioned	• Regularization factor design • Data-limited complexity $O(N^3)$

Outline

- Motivations, target applications & challenges
- Parametric regression
- Non-parametric regression
- Semi non-parametric regression
- Non-parametric regression for dynamical systems
- Conclusion and open problems

Representer theorem

$k(x, x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$: Mercel Kernel
1) $k(\cdot, \cdot)$ continous, $\mathcal{X}$: compact
2) symmetric: $k(x, x') = k(x', x)$
3) positive semidefinite: $K \in \mathbb{R}^{N \times N} \geq 0, \quad [K]_{i,j} = k(x_i, x_j), \forall x_i, \forall N,$

$\mu : \mathcal{X} \to \mathbb{R}^+$: measure function (sampling density)

$h(x) := T_{k,\mu}[g](x) := \int_{\mathcal{X}} g(x')k(x, x')d\mu(x')$: Hilbert-Schmidt integral operator
$h(x), g(x) \in \mathcal{L}^2(\mu)$

Since T is a linear operator → eigenvalues and eigenfunctions

$$T_{k,\mu}[\phi(x)] = \lambda\, \phi(x), \lambda \geq 0$$

Representer Theorem: Let $k(\cdot, \cdot)$ be a Mercer kernel on $\mathcal{X} \times \mathcal{X}$, $\lambda_\ell > 0 \quad \forall \ell$ and μ a non-degenerate measure. Then, $\{\phi_\ell\}_{\ell=1}^{+\infty}$ is an orthonormal basis in $\mathcal{L}^2(\mu)$ while the associated RKHS is

$$\mathcal{H}_K := \left\{ f(x) \in \mathcal{L}^2(\mu) \ \text{ s.t. } \ f(x) = \sum_{\ell=1}^{\infty} \alpha_\ell \phi_\ell(x) \text{ and } \sum_{e=1}^{\infty} \frac{\alpha_e^2}{\lambda_e} < +\infty \right\}$$

Map-building as least-squares regression

- Model class:

$$f(x) = \sum_{m=1}^{M} \theta_m g_m(x)$$

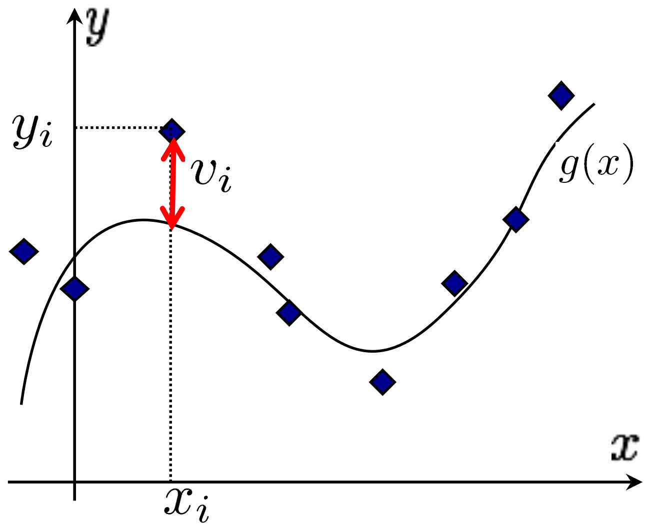
- Noisy measurements:

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

$$\begin{bmatrix} y(x_1) \\ y(x_2) \\ \vdots \end{bmatrix} = \begin{bmatrix} g_1(x_1) & \ldots & g_M(x_1) \\ \vdots & \vdots & \vdots \\ g_1(x_N) & \ldots & g_M(x_N) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_M \end{bmatrix} + \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix}$$

G_i^T

$$y = G\theta + v$$



- Goal: minimize sum of squares of residues

$$\hat{\theta} = \mathrm{argmin}_\theta \sum_{i=1}^{N} v_i^2$$

$$\hat{\theta} = \left(\sum_{i=1}^{N} G_i G_i^T\right)^{-1} \left(\sum_{i=1}^{N} G_i y_i\right) = \left(\frac{1}{N}\sum_{i=1}^{N} G_i G_i^T\right)^{-1} \left(\frac{1}{N}\sum_{i=1}^{N} G_i y_i\right)$$

- Xiao-Boyd-Lall, 2005
- Bolognani-Del Favero-Schenato-Varagnolo, 2010

Semi-parametric estimation

1st IDEA: Use first eigenfunctions as basis function for parametric estimation

$$f(x) = \sum_{\ell=1}^{+\infty} \alpha_\ell \phi_\ell(x)$$

$$y_i = \sum_{\ell=1}^{+\infty} \alpha_\ell \phi_\ell(x_i) = \underbrace{[\phi_1(x_i) \;\; \phi_2(x_i) \;\; \ldots]}_{G_i^T} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \end{bmatrix} + v_i$$

$$\widehat{\alpha} = \left(\mathrm{diag}(\tfrac{\sigma^2}{\lambda_\ell}) + \sum_{i=1}^N G_i G_i^T\right)^{-1} \left(\sum_{i=1}^N G_i y_i\right)$$

$$\widehat{\alpha}^E = \left(\mathrm{diag}(\tfrac{\sigma^2}{\lambda_\ell}) + \sum_{i=1}^N G_i^E (G_i^E)^T\right)^{-1} \left(\sum_{i=1}^N G_i^E y_i\right)$$

$$G_i^E = [\phi_1(x_i) \cdots \phi_E(x_i)]$$

$$(\text{intuition: } \alpha_i \approx 0, \text{for } i > E, \text{ therefore } \widehat{f}(x) \approx \widehat{f}^E(x))$$

Semi-parametric estimation (cont'd)

2st IDEA: Use orthonormality of eigenfunctions ϕ_n and i.i.d. sampling of x_i

$$\widehat{\alpha}^E = \left(\mathrm{diag}\left(\frac{\sigma^2}{N\lambda_\ell}\right) + \frac{1}{N}\sum_{i=1}^N G_i^E (G_i^E)^T\right)^{-1} \left(\frac{1}{N}\sum_{i=1}^N G_i^E y_i\right)$$

$$\left[\frac{1}{N}\sum_{i=1}^N G_i^E \left(G_i^E\right)^T\right]_{mn} = \frac{1}{N}\sum_{i=1}^N \phi_m\left(x_i\right)\phi_n\left(x_i\right)$$

$$\left[\frac{1}{N}\sum_{i=1}^N \phi_m(x_i)\phi_n(x_i)\right] \xrightarrow{N\to+\infty, x_i\sim\mu(x)} \int \phi_m(x)\phi_n(x)d\mu(x) = \delta_{mn}$$

$$\widehat{\alpha}^I(x) = \left(\mathrm{diag}\left(\frac{\sigma^2}{N\lambda_\ell}\right) + I\right)^{-1} \left(\frac{1}{N}\sum_{i=1}^N G_i^E y_i\right)$$

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Complexity of semi-parametric approaches

$$\widehat{f}(x) = \sum_{i=1}^{N} c_i k(x_i, x), \quad c = (K+I)^{-1} y, \quad [K]_{mn} = k(x_m, x_n)$$

$$\widehat{f}^E(x) = \sum_{i=1}^{E} \alpha_i^E \phi_i(x) \qquad \widehat{\alpha}^E = \left(\mathrm{diag}\left(\frac{\sigma^2}{N\lambda_\ell}\right) + \frac{1}{N}\sum_{i=1}^{N} G_i^E (G_i^E)^T\right)^{-1} \left(\frac{1}{N}\sum_{i=1}^{N} G_i^E y_i\right)$$

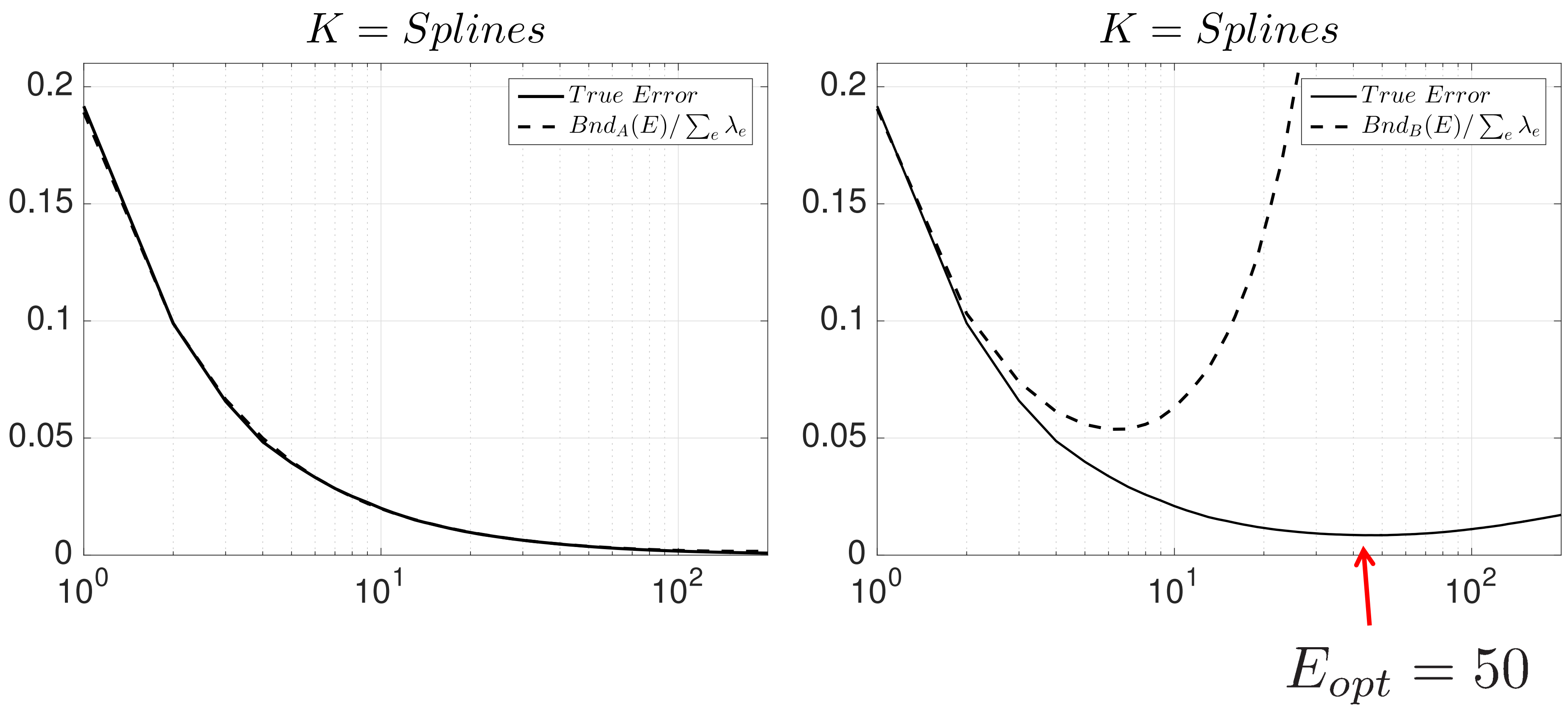
$$\widehat{f}^I(x) = \sum_{i=1}^{I} \alpha_i^E \phi_i(x) \qquad \widehat{\alpha}^I = \left(\mathrm{diag}\left(\frac{\sigma^2}{N\lambda_\ell}\right) + I\right)^{-1} \left(\frac{1}{N}\sum_{i=1}^{N} G_i^E y_i\right)$$

estimator	*comput. cost*	*commun. cost*	*memory cost*
$\widehat{f}(x)$	$O(N^3)$	$O(N)$	$O(N)$
$\widehat{f}^E(x)$	$O(E^3)$	$O(E^2)$	$O(E^2)$
$\widehat{f}^I(x)$	$O(E)$	$O(E)$	$O(E)$

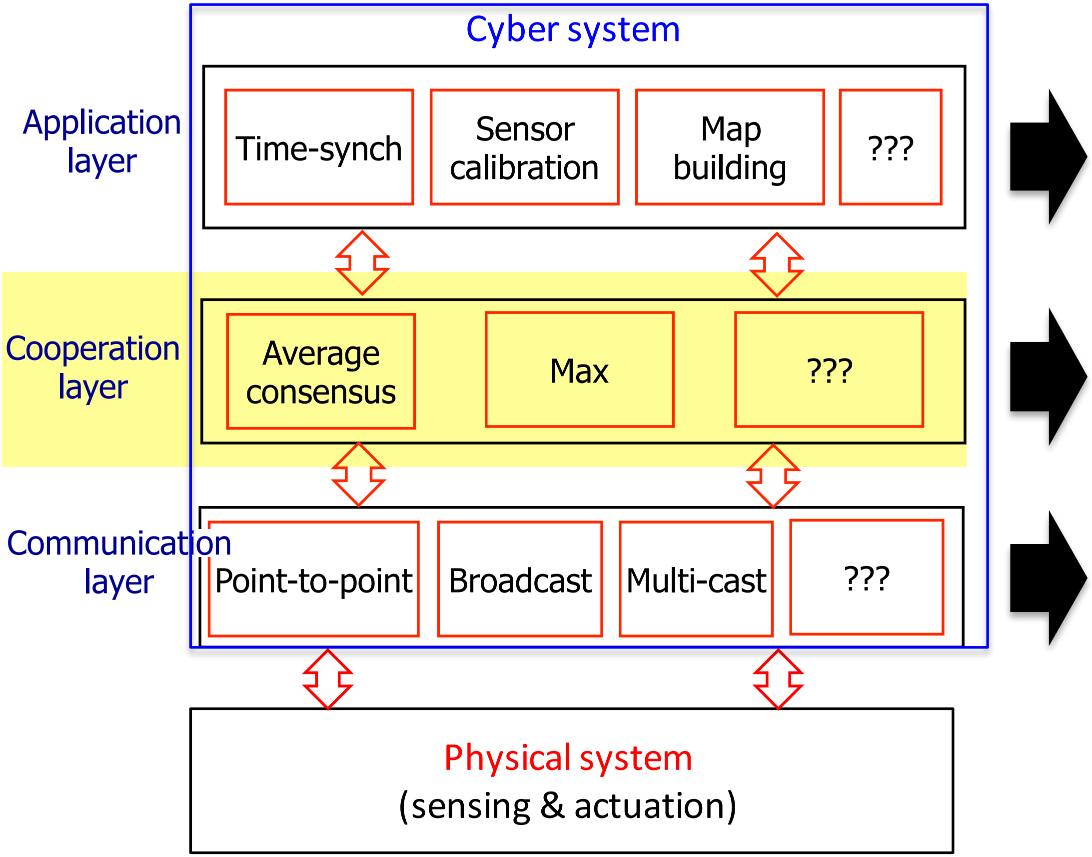
Performance of semi-parametric approaches

$$N = 10000$$

$\widehat{f}^E(x)$ $\widehat{f}^I(x)$



Current research agenda



- **Machine-learning** estimation
- Casting cooperative estimation & control as optimization problems

- **Distributed optimization** over unreliable communication

- WiFi-based protocols for control
- Adaptive **communication&control** rate via cross-layer desing

Interdisciplinary Approach

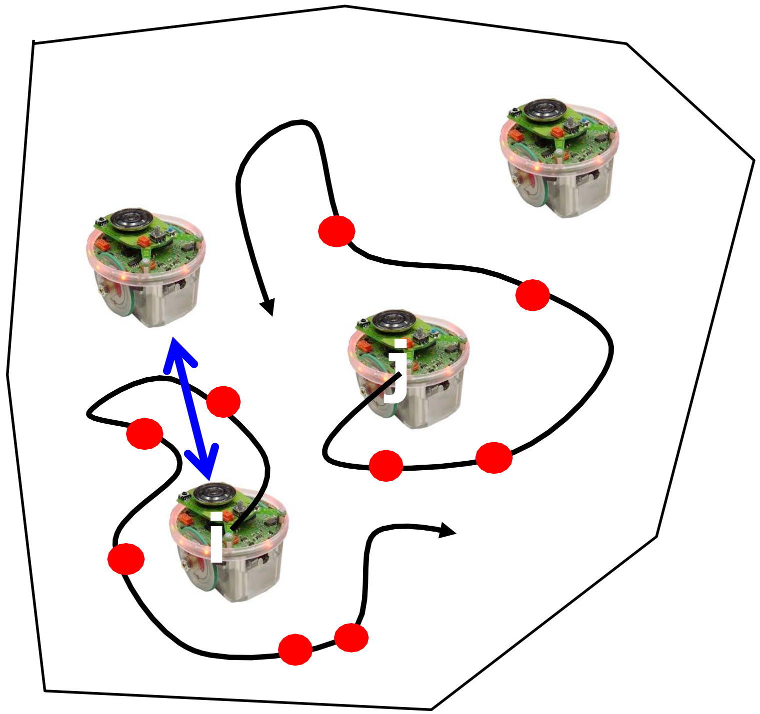
DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Outline

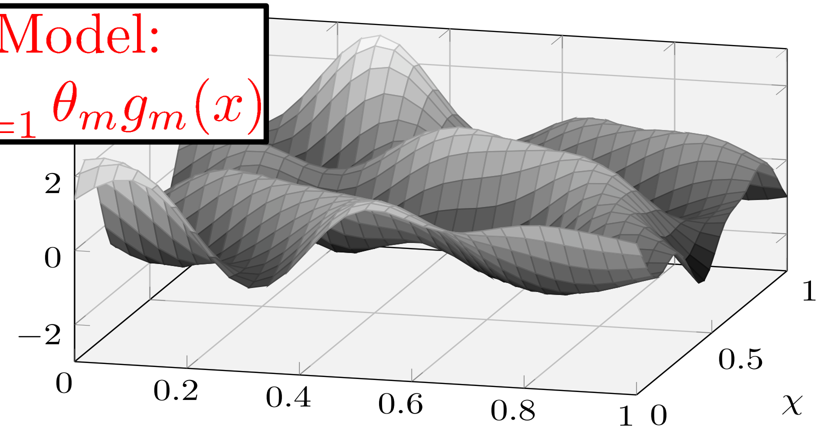
- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

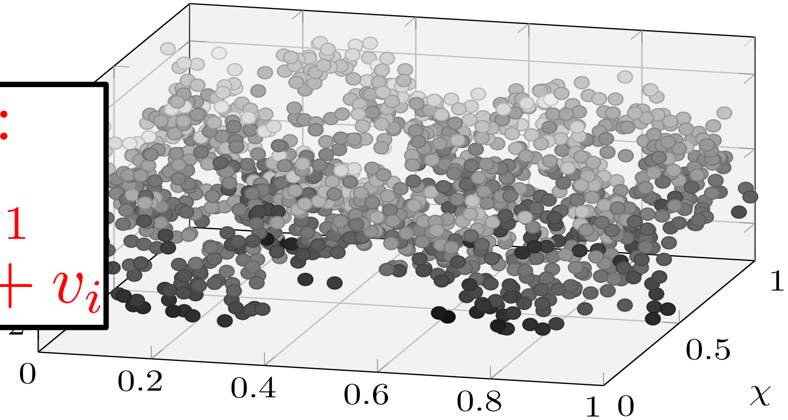
Robust regression

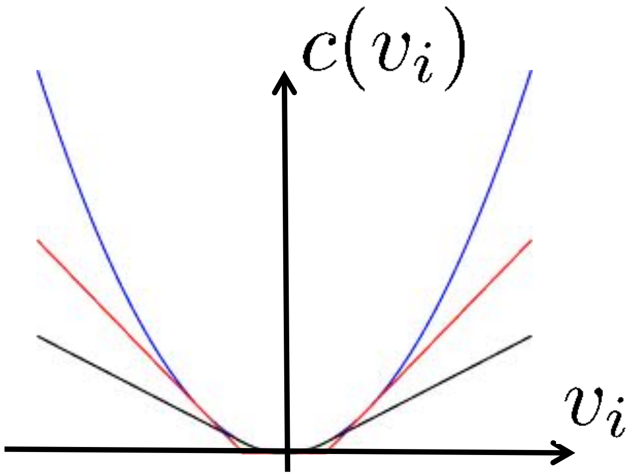


Parametric Model:
$g(x) = \sum_{m=1}^{M} \theta_m g_m(x)$



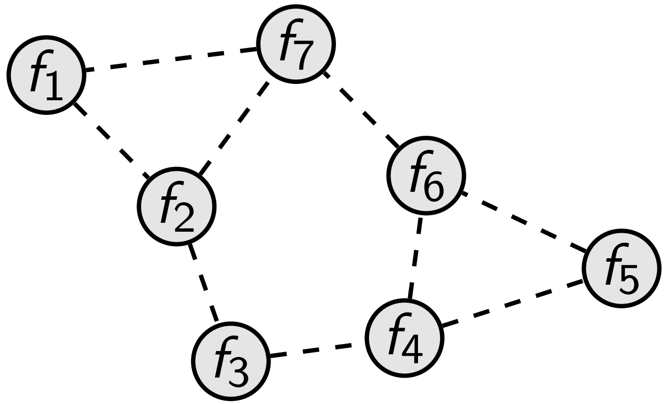
Noisy data:
$\{(x_i, y_i)\}_{i=1}^{N}$
$y_i = g(x_i) + v_i$





Goal:
$\min_\theta \sum_i \sqrt{1 + v_i^2} = \min_\theta \sum_i c(v_i) = \sum_i f_i(\theta)$

Distributed convex optimization: problem formulation



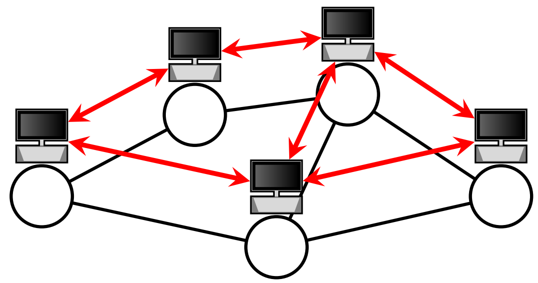
Assumption: neighbours cooperate to find minimizer of network cost:

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x), \quad x^* = \mathrm{argmin}_x f(x)$$

- Global estimation: $x \in \mathbb{R}^n$, each node wants $\hat{x}_i = x^*, \forall i = 1, \ldots, N$. Typically n independent of N: support vector machine, robotic map building.

Peer-to-peer convex optimization: state-of-the-art

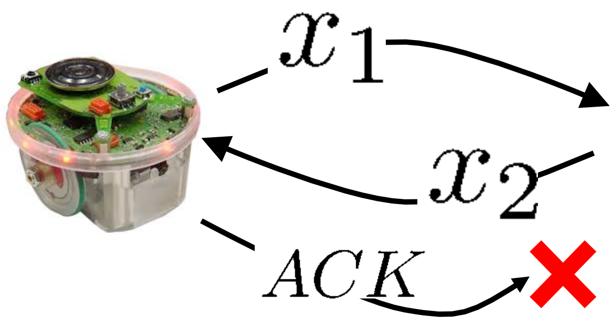
- Challenges:
  - Local communication
  - Time-varying graph
  - Lossy communication
  - Synchronization



- Main trends:
  - Distributed subgradient methods (Ozdaglar, Nedic,..)
    - No need for synchronization, OK for time-varying topologies
    - Slow convergence (sublinear) due to decreasing step-size
  - Alternating Direction Method of Multipliers (ADMM) (Bertsekas, Boyd,…)
    - Fast convergence (linear), applicable to many scenarios (de-facto standard)
    - Synchronous (very recent works for asynchronous past 2 years)
- No algorithms for lossy communication (that are we aware of)

Why lossy communication is relevant ?

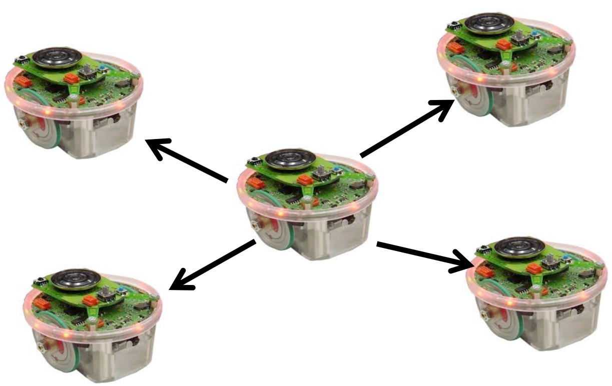
- Inconsistent information sets among agents: optimal cooperation hard (Witsenhousen's counterexample)



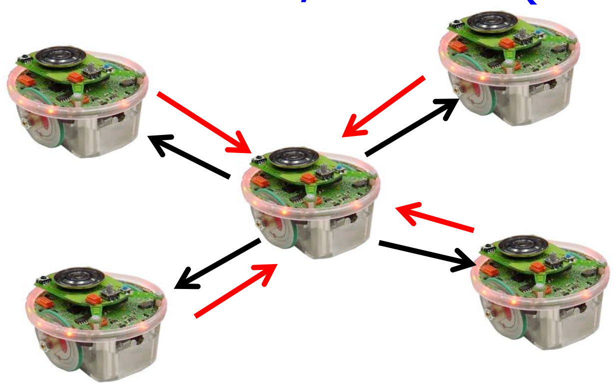
$$x_1^+ = (x_1 + x_2)/2$$
$$x_2^+ = (x_1 + x_2)/2$$

- ACK-based hard to implement and slows convergence

Broadcast w/o ACK = T



Broadcast w/ ACK = (N+1)T



Contribution

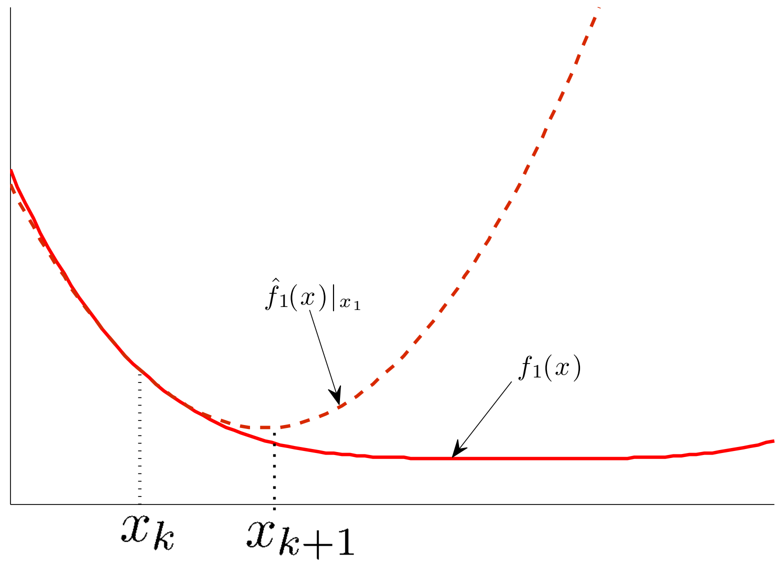
- Robust Newton-Raphson Consensus
  - Peer-to-peer
  - Asynchronous
  - Broadcast-based (no ACK needed)
  - Scalable (complexity/node is constant)
- Ideas: merging
  - Newton-Raphson consensus (our group, 2011)
  - Push-sum consensus (Benezit et al., 2010)
  - Robust ratio consensus (Dominguez-Garcia et al, 2011)

Newton-Raphson Consensus

$\min_x f(x), \quad x_k$ (current estimate)

Idea: approximate function $f(x)$ with a parabola

$$\widehat{f}(x) = \frac{1}{2}a(x-b)^2 + c$$



Match slope and curvature at point x_k:

$$\begin{aligned} f(x_k) &= \widehat{f}(x_k) = \tfrac{1}{2}a(x_k-b)^2 + c \\ f'(x_k) &= \widehat{f}'(x_k) = a(x_k-b) \\ f''(x_k) &= \widehat{f}''(x_k) = a \end{aligned}$$

Jump to the minimum:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \varepsilon(\nabla^2 f(\mathbf{x}_k))^{-1}\nabla f(\mathbf{x}_k)$$

Newton-Raphson Consensus

$$\min_x \sum_i f_i(x), \quad \{x_k^i\}_{i=1}^N \text{ (current estimates)}$$

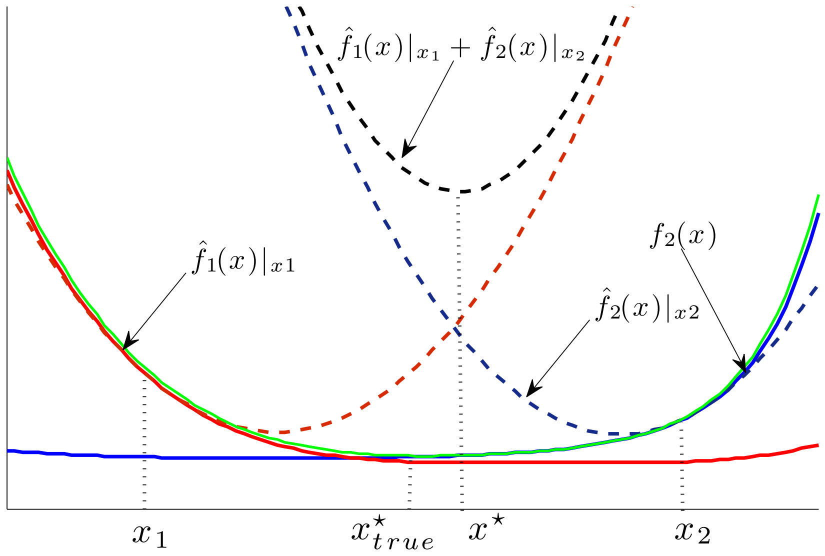
Approximate **each** $f_i(x)$ with a parabola

$$\widehat{f}_i(x) = \frac{1}{2} a_i (x - b_i)^2 + c_i$$

Match slope and curvature at point x_i

$$f_i'(x_i) = \widehat{f}_i'(x_i) = a_i(x_i - b_i)$$
$$f_i''(x_i) = \widehat{f}_i''(x_i) = a_i \quad \Rightarrow$$

Jump to the minimum of $\widehat{f}(x) = \sum_i \widehat{f}_i(x)$



$$x_i^+ = x^* = \frac{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)x_i - f_i'(x_i)}{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)}$$

if all points are the same, i.e. $x_i = x \ \forall i$, then:

$$x_i^+ = x^+ = x - \frac{\frac{1}{N}\sum_{i=1}^N f_i'(x_i)}{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)} = x - \frac{f'(x)}{f''(x)}$$

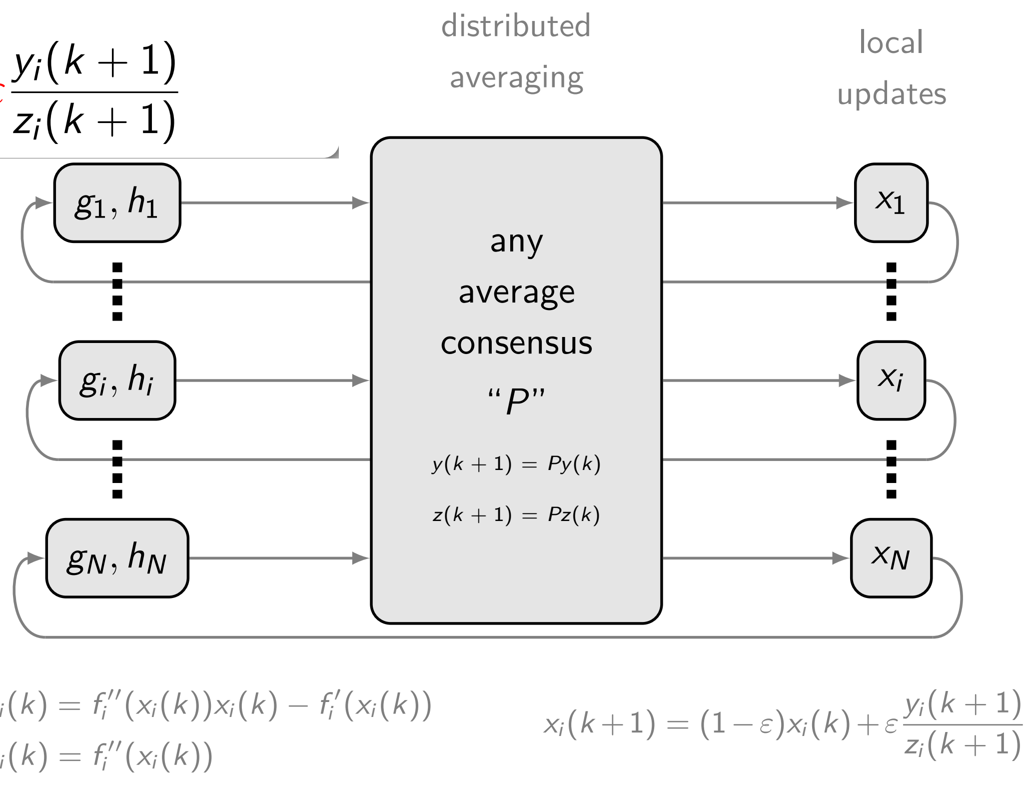
Newton-Raphson Consensus

Algorithm

1. define local variables:
   - $g_i(k) := f_i''(x_i(k))x_i(k) - f_i'(x_i(k)), \quad g_i(-1) = 0, \quad y_i(0) = 0$
   - $h_i(k) := f_i''(x_i(k)), \quad h_i(-1) = 0, \quad z_i(0)$
2. run 2 average consensus (P doubly stochastic):
   - $\mathbf{y}(k+1) = P\mathbf{y}(k)$
   - $\mathbf{z}(k+1) = P\mathbf{z}(k)$
3. locally compute $x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \dfrac{y_i(k+1)}{z_i(k+1)}$

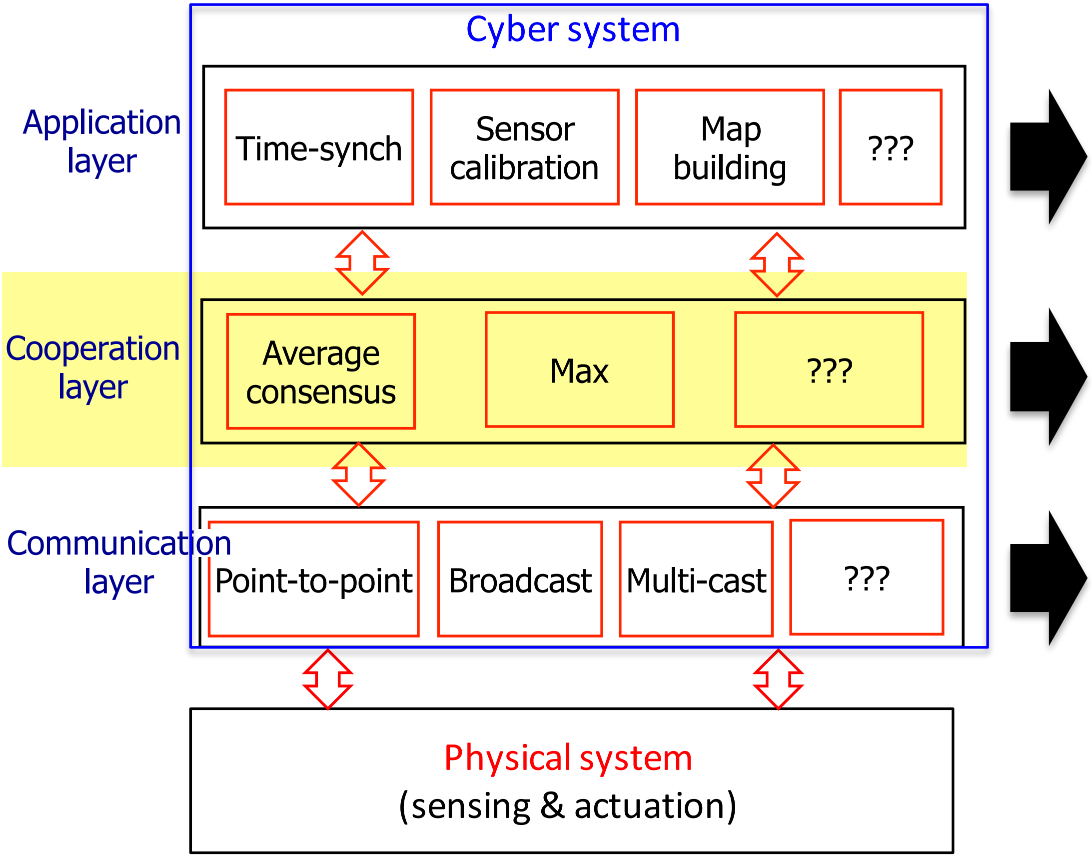
$$x^* = \frac{\frac{1}{N}\sum_{i=1}^{N} f_i''(x_i)x_i - f_i'(x_i)}{\frac{1}{N}\sum_{i=1}^{N} f_i''(x_i)}$$

Currently re-discovered by other groups:
Nedic, Wei, Na, Scutari



Simulations: broadcast based map building

Current research agenda



- **Machine-learning** estimation
- Casting cooperative estimation & control as optimization problems

- **Distributed optimization** over unreliable communication

- WiFi-based protocols for control
- Adaptive **communication&control** rate via cross-layer desing

Interdisciplinary Approach

Outline

- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Sampling time vs packet loss

- Wi-Fi (802.11)
  - Multiple bit-rate to choose
  - Higher bit-rate = smaller sampling time, higher losses
  - Currently optimize throughput

- Time-critical applications
  - Both sampling time, delay and packet loss degrade performance and all depends on bit-rate
  - Is throughput a good metric for control?
  - Is packet loss as bad as delay?
  - How control deal with dynamic sampling time and packet losses?

Non-intuitive answers

Let us consider:

$$dx(t) = ax(t)dt + u(t)dt + dw(t), \quad E[(dw(t))^2] = 1$$

Sampling at period T=1/R, R=packet-rate

$$x_{k+1} = (1 + aT)x_k + Tu_k + \sqrt{T}w_k, \quad E[w_k^2] = 1, \quad x_k = x(kT)$$

Lossy transmission: $P[\gamma_k=0]=\gamma$

$$y_k = \gamma_k x_k$$

Dead-beat controller:

$$u_k = -\frac{1}{T}(1 + aT)x_k$$

Dead-beat controller:

$$x_{k+1} = (1 - \gamma_k)(1 + aT)x_k + \sqrt{T}w_k, \quad E[w_k^2] = 1$$

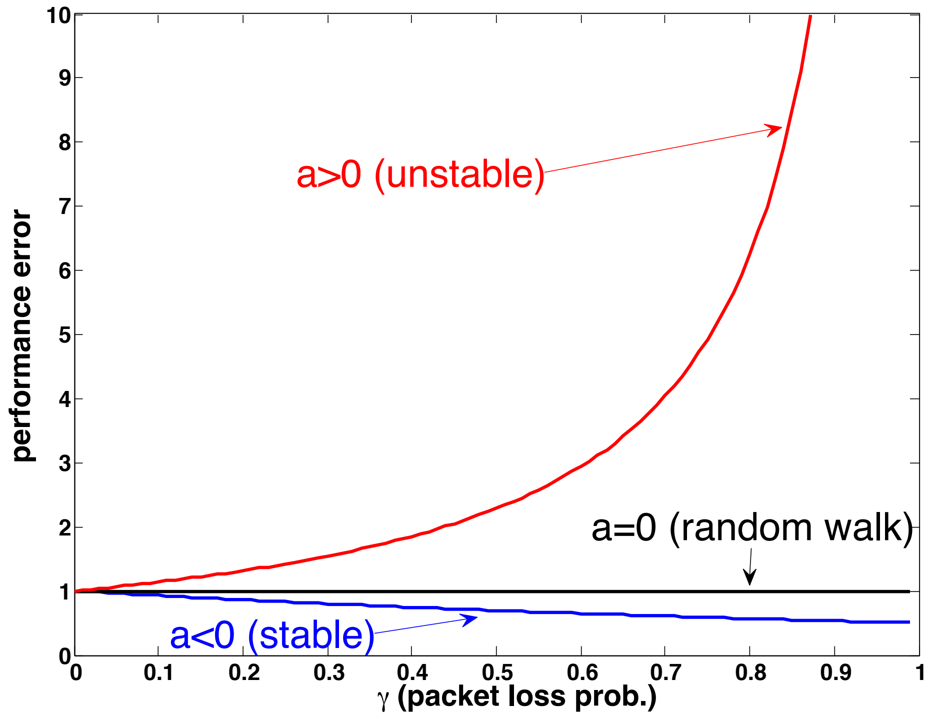
Non-intuitive answers: constant throughput

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Average performance (mean square error)

$$p := \lim_{k\to\infty} E[x_k^2] \Longrightarrow p = \frac{T}{1-\gamma(1+aT)^2}$$

Two protocols with same throughput: $\frac{1-\gamma}{T} = \text{cost.}$



Best protocol depends on stability of controlled system

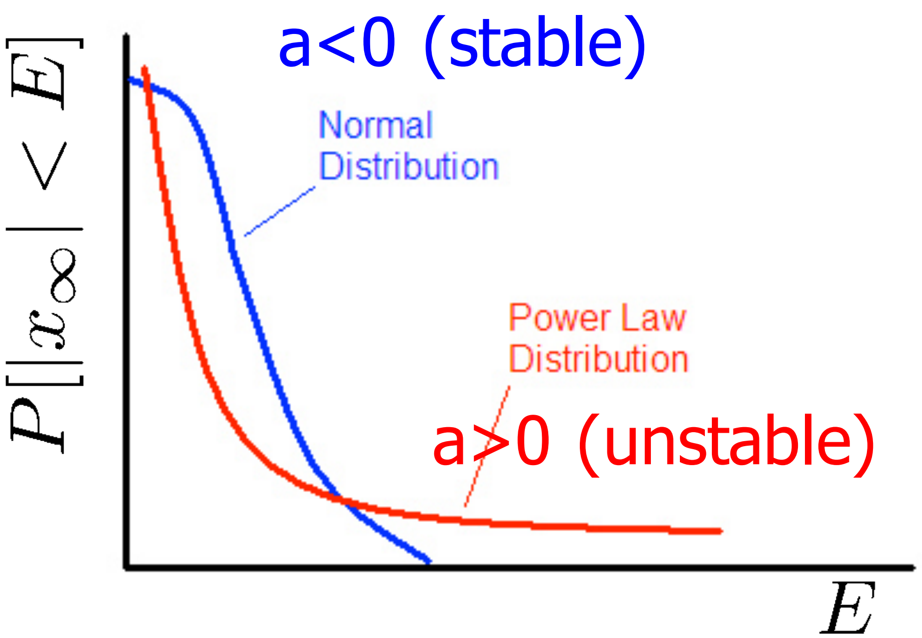
Non-intuitive answers: heavy tail

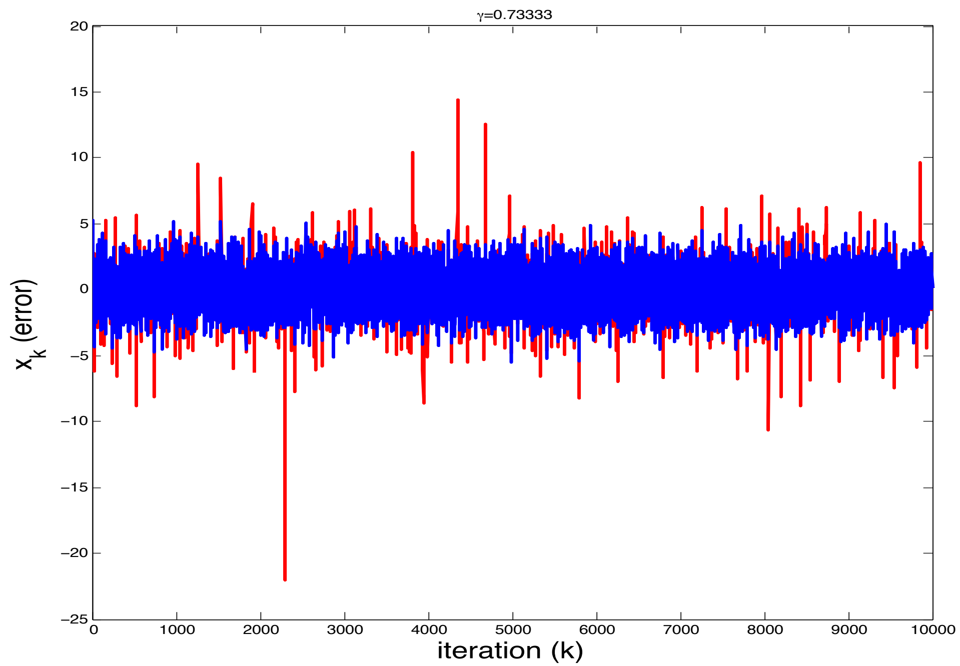
Average performance (mean square error MSE)

$$p := \lim_{k\to\infty} E[x_k^2] \Longrightarrow p = \frac{T}{1-\gamma(1+aT)^2}$$

Two protocols with same MSE:

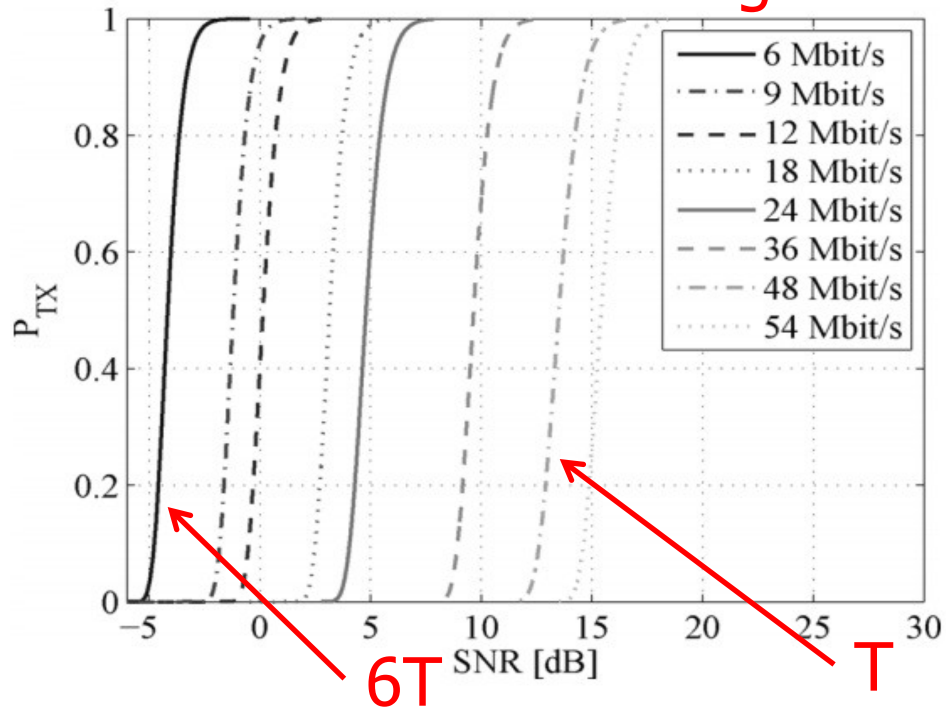
$$\frac{T}{1-\gamma(1+aT)^2} = \text{cost.}$$

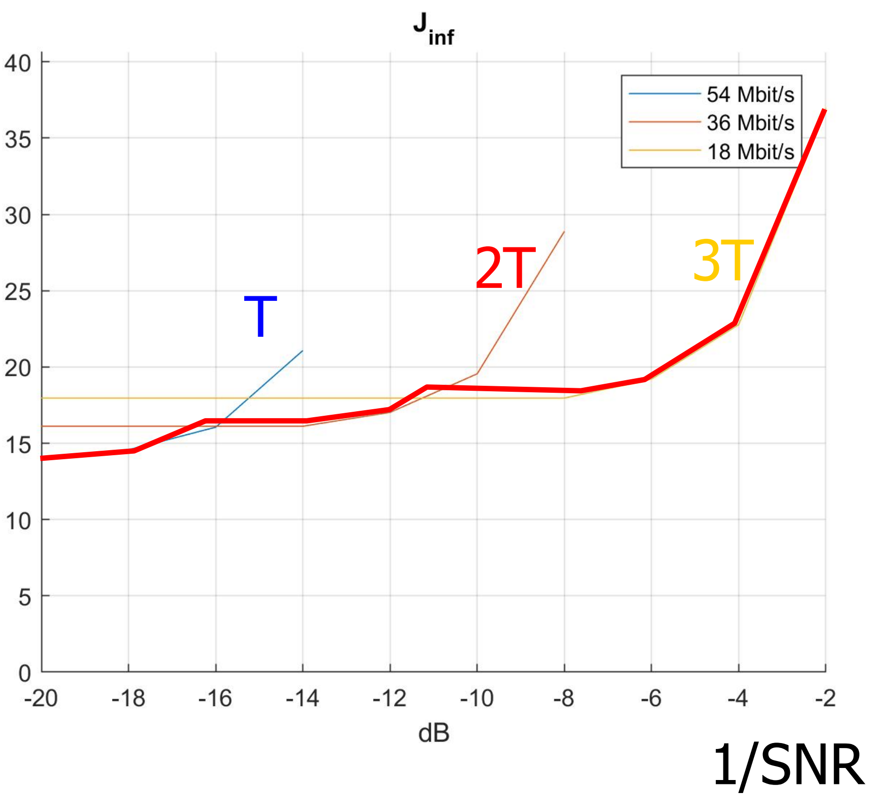




Adaptive rate selection for control

Wi-Fi 802.11g rates





$$J_M(u) = \mathbb{E}\left[\frac{1}{M}\int_0^M \left(x^T(t)Wx(t) + u^T(t)Uu(t)\right)dt\right.$$

$$J_K^* = \min_{\{u_k\}_{k=1}^{K-1}} J_K(\{u_k\}_{k=0}^{K-1}),$$

LQG control

$$J^*(1/SNR) = \min_i J_i^*(1/SNR)$$

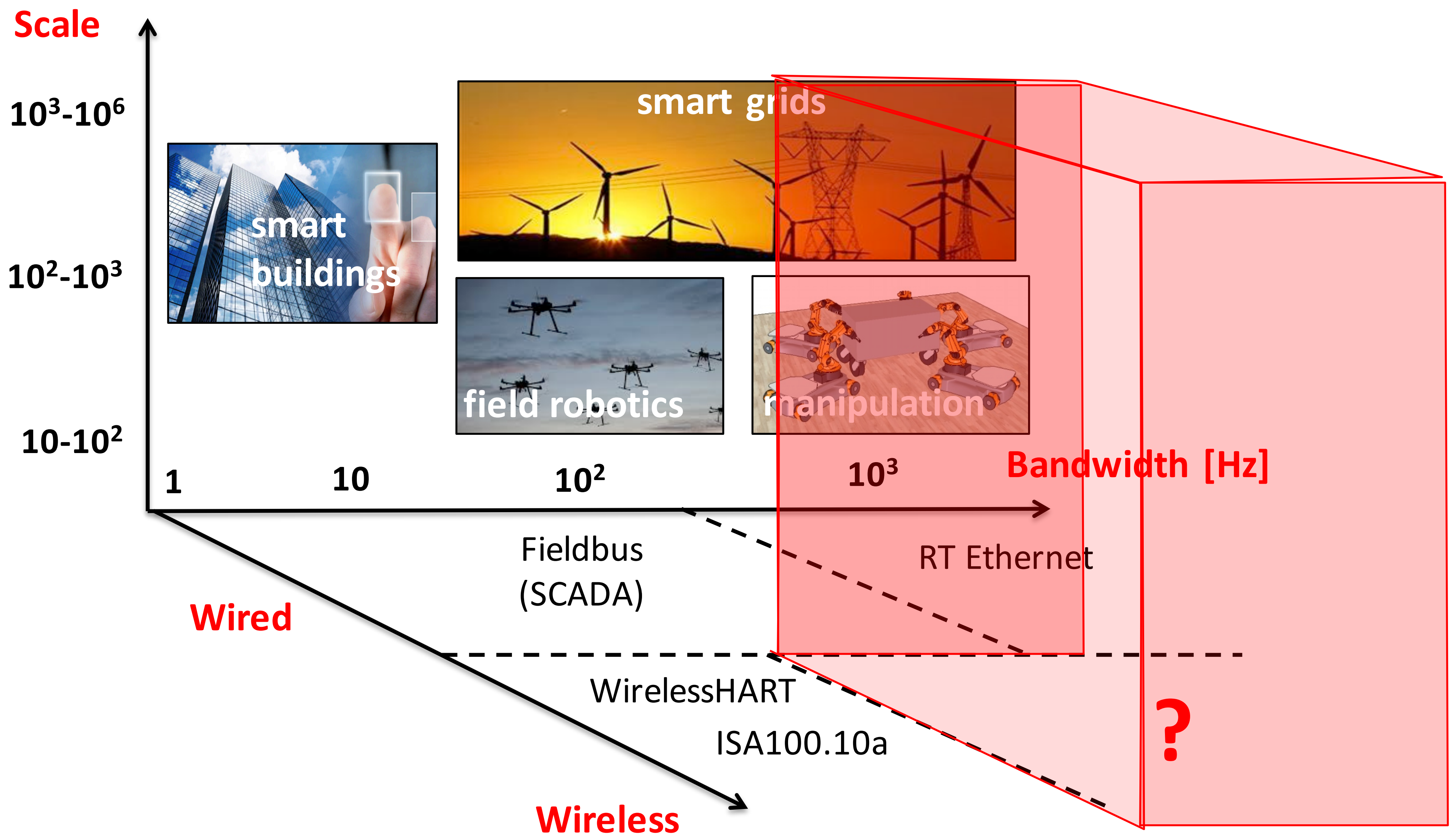
$$i^*(1/SNR) = \text{argmin}_i J_i^*(1/SNR)$$

Outline

- Motivations, target applications & challenges
- Map building: non-parametric regression
- Distributed optimization over lossy networks
- Adaptive Wi-Fi rate selection for control
- Future research agenda and conclusions

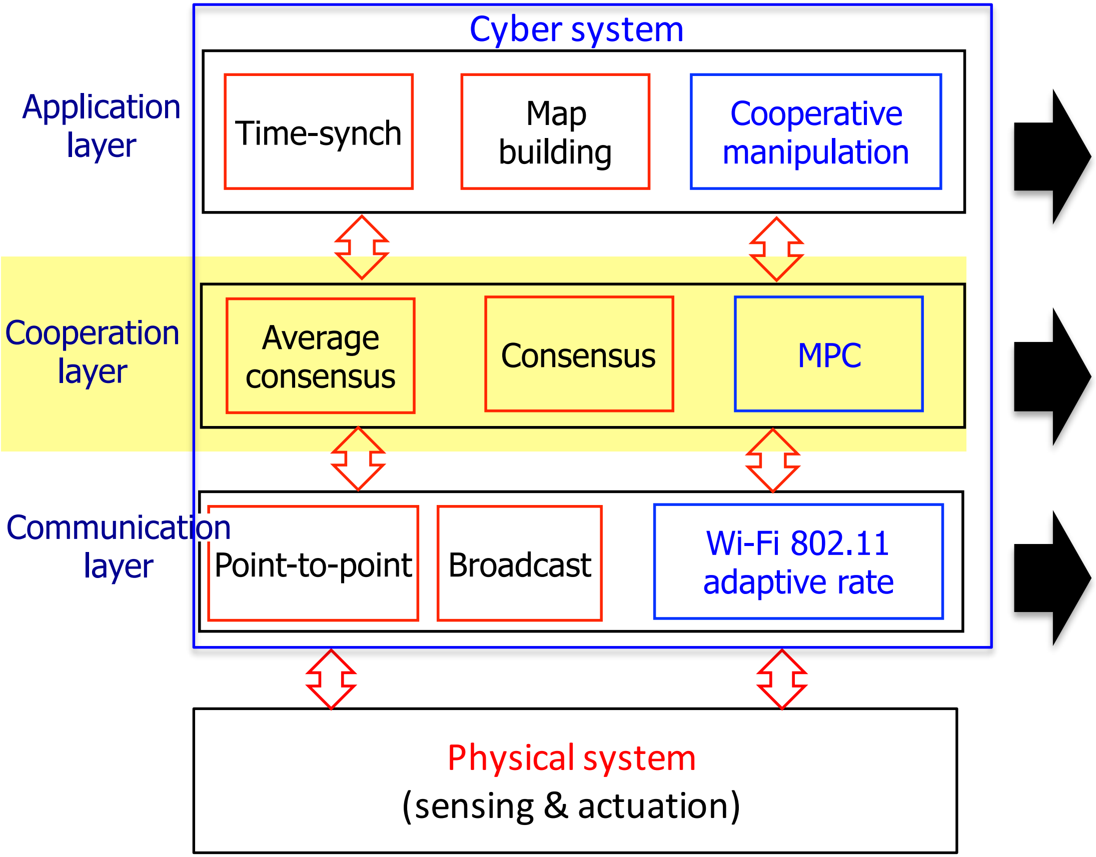
The challenge cube for time-critical smart systems

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA



DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Future research agenda



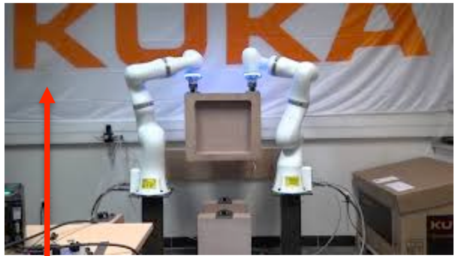
- **Machine-learning:** Gaussian Process-based learning
- **Distributed optimization:** MPC over lossy communication
- **Communication:** Wi-Fi 802.11 protocols for multi-agent control systems

Proof-of-concept: UAV manipulation over wireless

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Bandwidth [Hz]

1000 Hz

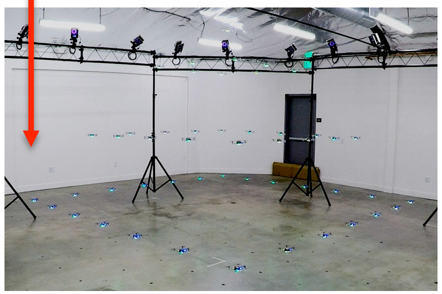


Manipulation bandwidth requirement: **>1kHz** using PID or classic control. Today only via wired communication.

100 Hz

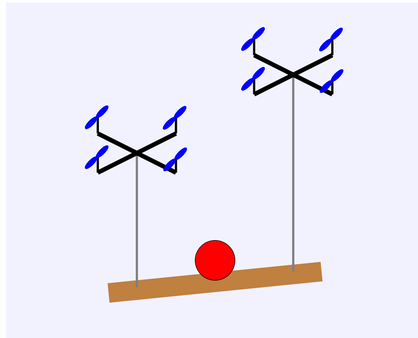
2-order of magnitude gap!!

10 Hz



Reliable WI-FI for control (no packet loss, constant delay): **<20Hz**. Today only for formation control.

Cooperative manipulation at **200Hz** via distributed MPC control



WI-FI for control up to **200Hz** via 802.11 real-time rate adaptation

Today's state-of-the-art

Tomorrow's Project goal

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Conclusions & open problems

- Smart Multi-agent control systems over wireless: currently more open questions than answers
- Need to look at realistic assumptions (in particular communication)
- Cooperative UAV manipulation over Wi-Fi in unstructured environments (outdoor): pristine area

References

- Map-building
  - D. Varagnolo, G. Pillonetto, L. Schenato. **Distributed multi-agent Gaussian regression via Karhunen-Loève expansions.** [under review IEEE PAMI] available on *Arxiv*
  - M. Todescato, A. Carron, R. Carli, G. Pillonetto, L. Schenato. **Efficient Spatio-Temporal Gaussian Regression via Kalman Filtering.** [under review Machine Learning Journal] *available on Arxiv*

- Convex optimization with lossy communication
  - N. Bof, R. Carli, G. Notarstefano, L. Schenato, D. Varagnolo. **Newton-Raphson Consensus under asynchronous and lossy communications for peer-to-peer networks.** [under review *IEEE Trans. Automatic Control], available on Arxiv*
  - M. Todescato, N. Bof, G. Cavraro, R. Carli, L. Schenato. **Generalized gradient optimization over lossy networks for partition-based estimation** *available on Arxiv*

- Adaptive rate selection for control systems:
  - S. Dey, L. Schenato. **Heavy-tails in Kalman filtering with packet losses: confidence bounds vs second moment stability.** *ECC'18 (submitted)*

Q&A

Thank you