Adaptive & Optimization tools for the Rapid, Agile Deployment & Operation of Large-Scale Systems (LSS)

Application to Robotics Swarms, Energy Positive Buildings and Traffic Control Systems

Elias B. Kosmatopoulos & Kostas Aboudolas

Dept. Of ECE, Democritus University of Thrace

Design, Deployment & Operation of LSS

- LSS involving real-time operations (e.g. control systems, sensor or communication networks):
  - Optimal design: typically NP-complete
  - Uncertainties wrt environment the LSS is operating
  - Small-, medium- and long-term variations of exogenous factors
  - Need for frequent re-design and re-configuration
- Examples
  - Swarms of Flying Robots (mobile sensor/communication networks) operating in unknown environments (FP7 sFLY project)
  - Operation of Energy Positive Buildings (EPBs: FP7 PEBBLE project)
  - Design and operation of Traffic Control Systems (FP4-FP6 projects SMART NETS, RHYTHM, EURAMP, COOPERS, …)

Problem 1: Adaptive Optimization of LSS

- LSS operating in uncertain (or, unknown), varying environments
  - Typically, the LSS is designed using
    - sub-optimal (e.g. based on linearized models) techniques
    - Based on inaccurate models for the LSS
  - Need for on-line calibration of a large number of «control parameters» in an attempt to optimize LSS's performance
- Examples
  - Swarms of Robots performing optimal surveillance coverage:
    - Move the robots so that a coverage criterion is optimized; Criterion is a function of robots' positions/orientations + unknown terrain morphology
  - Energy Positive Buildings & Traffic Control Systems: LSS decision making mechanism (sub-optimal) of the form
    - **u=K(Wy)**; **u** decisions, **y** sensor measurements; **W** parameters to be optimized
    - Highly uncertain system dynamics
    - System performance is an unknown «function» wrt **W**

The Congitive-based Adaptive Optimization (CAO) algorithm

- Problem formulation:

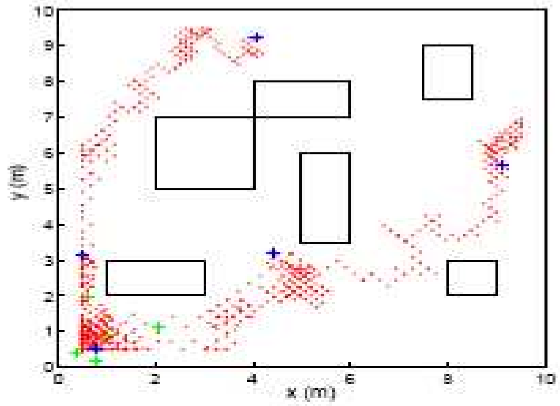
Optimize J(W, x)

  - **J(.)** is unknown but available for measurement at each iteration.
  - **W** is a vector with the parameters to be optimized
  - **x** are external factors (e.g. traffic demand, weather conditions or occupants' behavior)
- The CAO algorithm:
  - Form (at each iteration) an approximator **Ψ(W, ж)**, where **ж** is an estimate/prediction of **x** and train the approximator (using e.g. standard least-squares techniques)
  - Select randomly many candidate perturbations $\Delta\mathbf{W}^{(1)},\ldots,\Delta\mathbf{W}^{(K)}$
  - Choose the best $\mathbf{\Psi}(\mathbf{W}+\Delta\mathbf{W}^{(i)}, ж)$
- **Convergence theorem**: Under some mild regularity conditions and provided the random choice of the candidate perturbations is appropriately done,

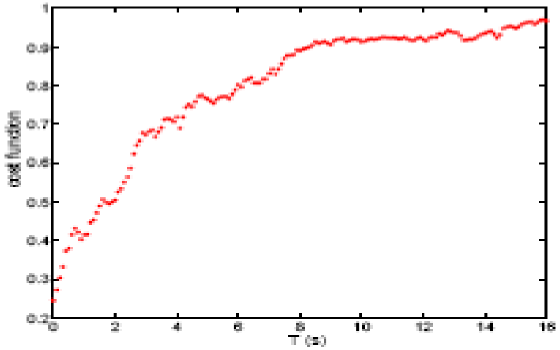
W→ W* +O(|x- ж|), w.p.1, W* is a local minimum of E[J(.)]

- Outperforms significantly alternative algorithms (e.g. SPSA by J. Spall)
- Most importantly, it gurantees safe convergence
- E. Kosmatopoulos *et al. IEEE Transactions on Control Systems Technology*, 2007; E. Kosmatopoulos, *Automatica*, 2009; E. Kosmatopoulos & A. Kouvelas, *IEEE Transactions on Neural Networks*, 2009.

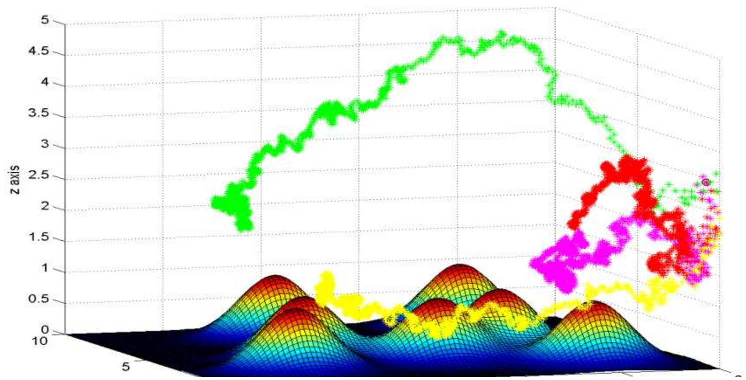
Multi-robot Surveillance Coverage

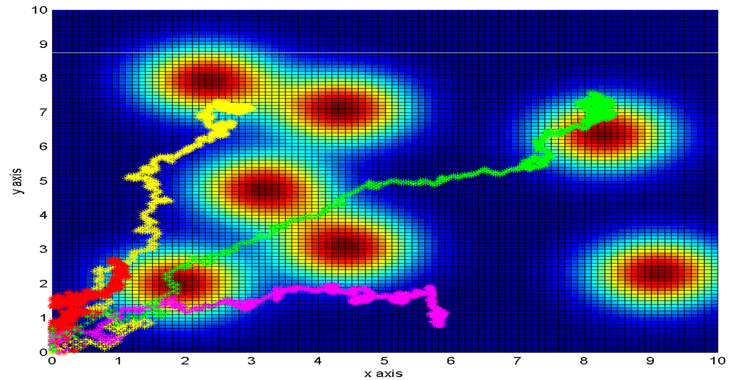


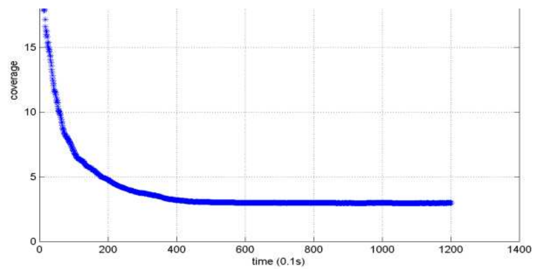
(a)



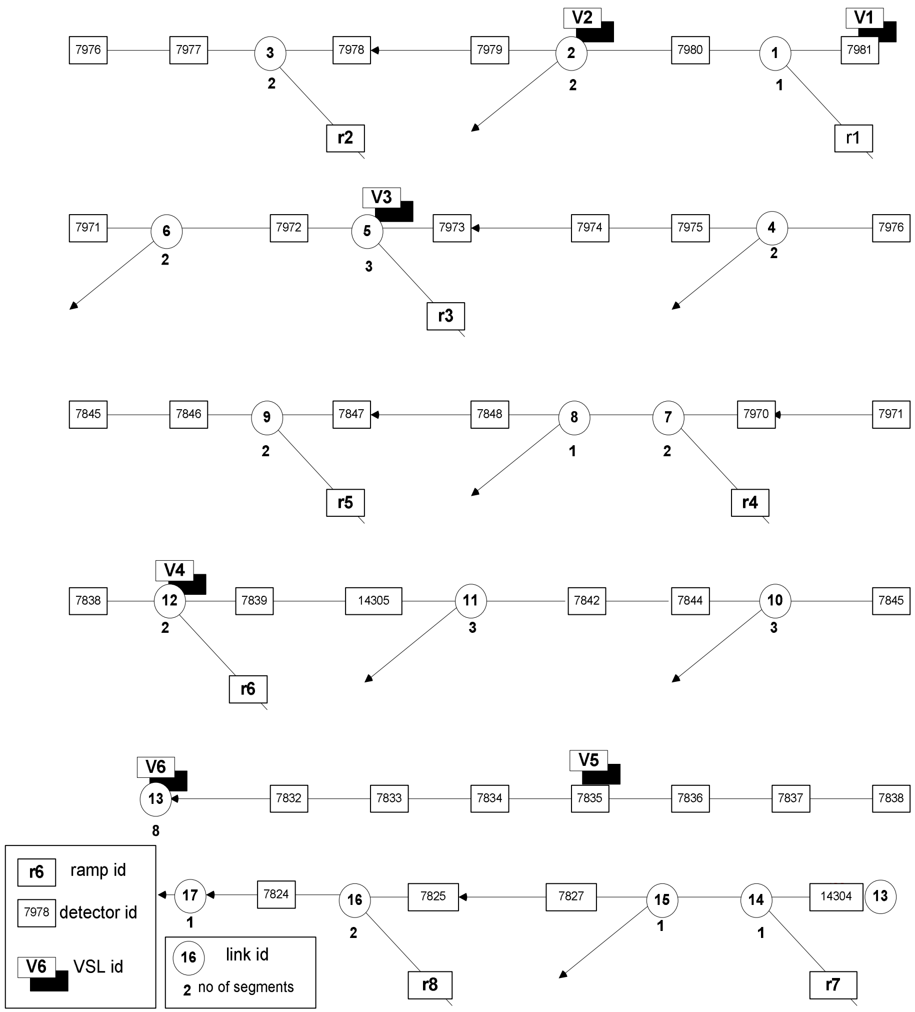
(b)



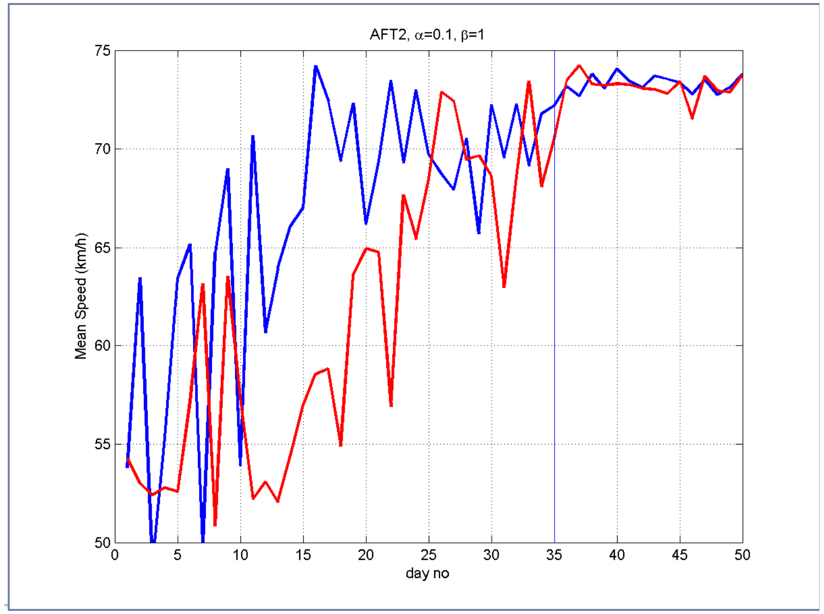




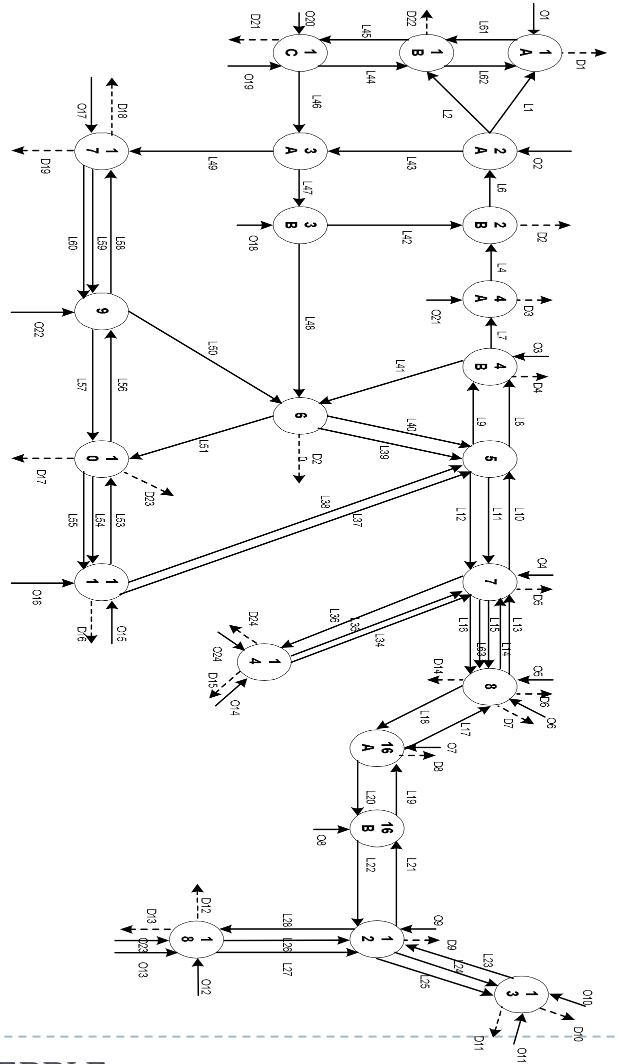
Traffic Control Systems (Motorway VSL system)



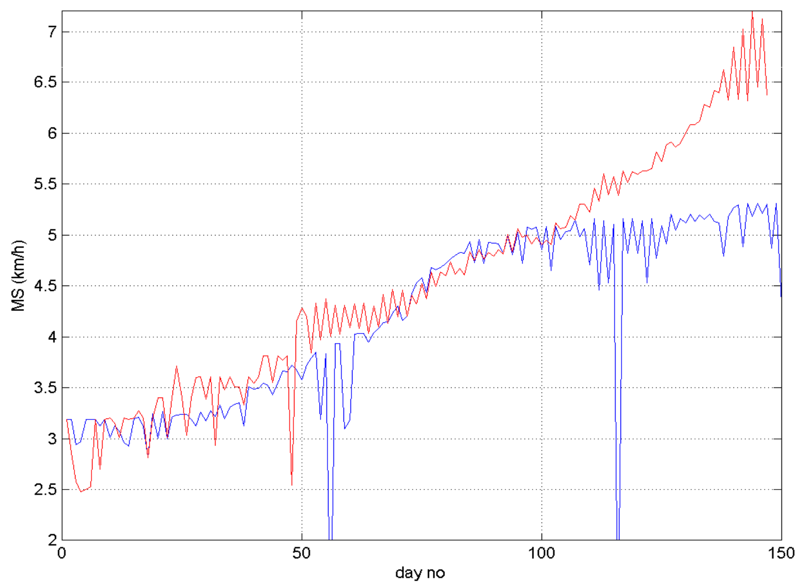
- 20 km-long motorway (Monash, Melbourne, Australia)
- 6 Variable Speed Limits
- Threshold-based switching logic
- Thresholds are optimized using CAO (24 thresholds)
- Optimization Criterion = Mean Speed



Traffic Control Systems (Chania Urban Traffic Network)



- •20 Junctions
- •~550 control parameters
- •Control-based logic
- •Optimization Criterion = Mean Speed
- •Total failure of SPSA



Problem 2: Practically Implementable LSS Design (Full-knowledge of LSS dynamics)

LSS dynamics

$$\dot{x} = f(x) + g(x)u + g_{\omega}(x)\omega$$

$$y = h(x)$$

$x \in \Re^n$	states
$u \in \Re^m$	tuneable or control inputs
$y \in \Re^k$	sensor measurements
f, g, h, g_{ω}	nonlinear functions
ω	disturbances (for simplicity we assume no sensor noise)

Examples: Robotic Swarms performing mapping, exploration and target tracking, controllers for energy positive bulidings, traffic control systems

Optimal LSS Design

Estimator Structure

$$\dot{\hat{x}} = f(\hat{x}) + g(\hat{x})u + u_0$$

$$\hat{y} = h(\hat{x})$$

u_0 correction signal

Criterion (Constraints may be also added)

$$\min_{(u(s),u_0(s),s\in[0,\infty))} J(\overline{x}(0),\hat{\overline{x}}(0))$$

$$J(\overline{x}(0),\hat{\overline{x}}(0)) = E\left[\int_0^\infty \left(|u(s)|_R^2 + |u_0(s)|_{R_0}^2 + |\overline{x}(s) - \hat{\overline{x}}(s)|_Q^2 \right) ds \right]$$

$R, R_0, Q \succ 0$ $\quad \overline{x} = vec(x, y)$ $\quad \hat{\overline{x}} = vec(\hat{x}, y_r)$

y_r desired level for sensor measurements

Examples

- Sensor Networks
  - **y**, $\mathbf{y_r}$ do not appear in the optimization criterion (no desired level for sensor measurements)
  - **R=0** and **u** is given, passive sensing; otherwise active sensing
  - Special case: passive sensing with optimal sensor design problem
- Optimal Control LSS design in most other cases (under appropriate transformations if needed)

Approach

- Approximate the «Optimal-Cost-to-Go» function $J(\bar{x}(t),\hat{\bar{x}}(t)) = E\left[\int_0^\infty \left(|u(s)|_R^2 + |u_0(s)|_{R_0}^2 + |\bar{x}(s) - \hat{\bar{x}}(s)|_Q^2 \right) ds \right]$ using a SOS polynomial
- Approximate the Optimal Actions using Piecewise Linear Functions
- Employ the Hamilton-Jacobi-Bellman equation and the property of the «optimal-cost-to-go» function that is positive definite

$$V\left(\bar{x},\hat{\bar{x}}\right) = z^{\tau}(\bar{x},\hat{\bar{x}})Pz(\bar{x},\hat{\bar{x}})$$

$$u^{*}(t) = \sum_{j=1}^{R(L)} \theta_c^{(j)}\phi_j(Y)(y - y_r)$$

$$u_0^{*}(t) = \sum_{j=1}^{R(L)} \theta_0^{(j)}\phi_j(Y)(y - h(\hat{x}(t)))$$

Approach

- Choose many random points for the system/estimator states and make sure that the HJB is approximately satisfied and the «optimal-cost-to-go-function» approximation is positive definite.

$$\min_{\theta_c,\theta_0,P,\mu_i} \quad \sum_{i=1}^{\aleph} \mu_\iota$$

$$\mu_i \quad \geq \quad V_x^{[i]\tau} f\left(x^{[i]}\right) + V_{\hat{x}}^{[i]\tau} f\left(\hat{x}^{[i]}\right) + V_y^{[i]\tau} h_x(x) f(x)$$

$$+\frac{1}{2} tr\left\{g_\omega(x) V_{xx} g_\omega(x^{[i]})\right\} + \frac{1}{2} tr\left\{[h_x(x^{[i]}) g_\omega(x^{[i]})]^\tau V_{yy} h_x(x^{[i]}) g_\omega(x^{[i]})\right\}$$

$$+\left[V_x g(x^{[i]}) + V_{\hat{x}} g(\hat{x}^{[i]})\right] k_c(Y)(y - y_r) + V_{\hat{x}} k_o(Y)[y - h(\hat{x}^{[i]})]$$

$$\mu_i \quad \geq \quad 0$$

$$V^{[i]} \quad \geq \quad -\varepsilon_1 \left|\overline{x}^{[i]} - \hat{\overline{x}}^{[i]}\right|^2, \quad V^{[i]} \leq -\varepsilon_2 \left|\overline{x}^{[i]} - \hat{\overline{x}}^{[i]}\right|^2$$

$$P \quad \geq \quad 0$$

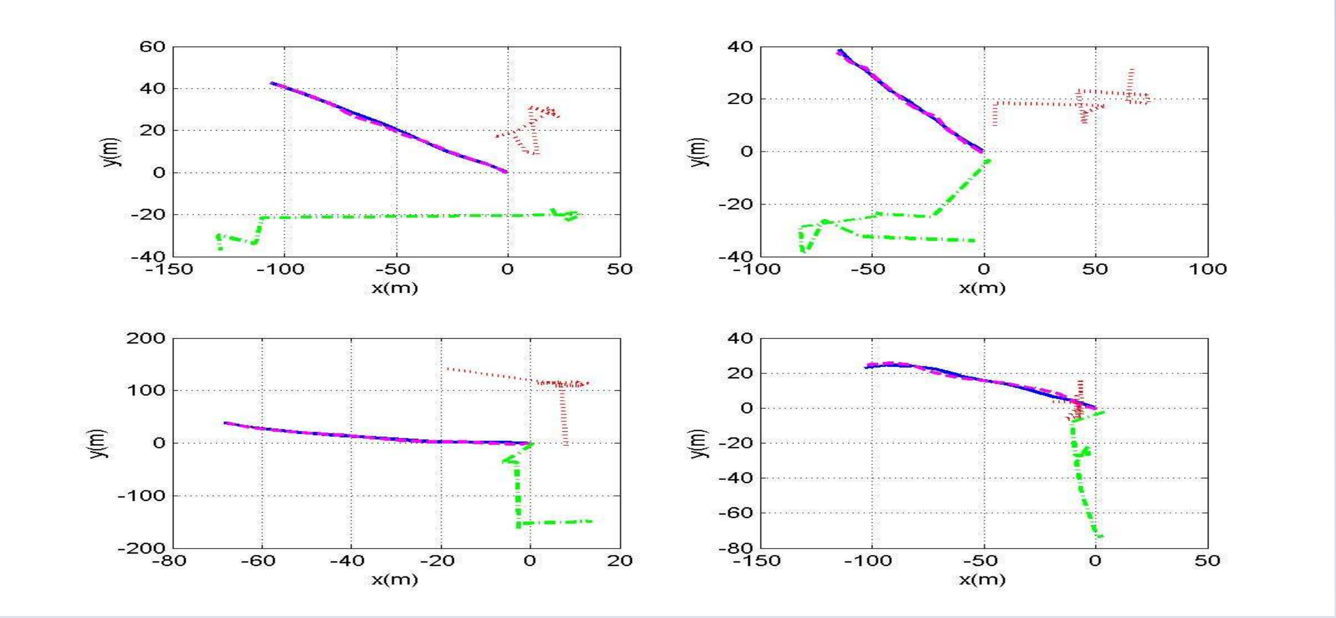
Approach

- Convex Optimization Problem!
- Stability is preserved (optimal-cost-to-go function is a Lyapunov function)
- The approximate solution can be made arbitrarily close to the optimal solution
- Heavy computations are made off-line
- Real-time requirements similar to those of linear mechanisms

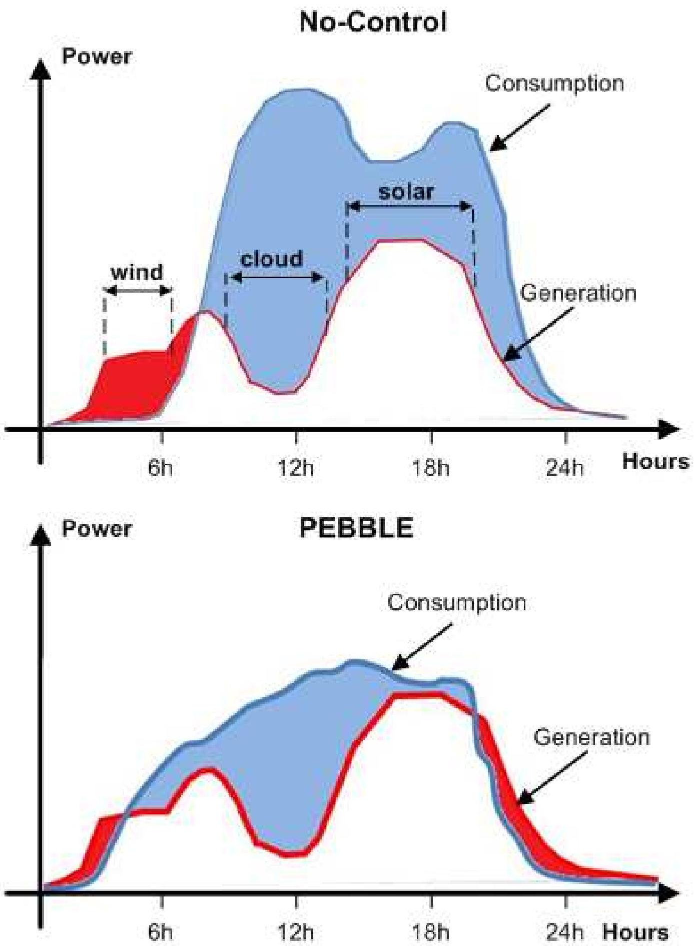
E. Kosmatopoulos, L. Doitsidis, K. Aboudolas. *IEEE IROS 2010* and FP7 sFLY report.

Applications: Multi-Robot Target Tracking

- 3D target tracking (2 flying robots)
  - Linear Estimator + Covariance Minimizer fails
  - Proposed solution achieves an accuracy that is 3 times the sensor noise (27 switching elements).
- 2D target tracking (2 mobile robots; 16 switching elements)



Applications: Energy-Positive Buildings (EPBs)



Area under consumption curve: energy required for building operation.
Area under generation curve: energy available by installed renewable energy sources.
Red Area: Surplus energy available.
Blue Area: Energy purchased from the grid.

Current View: "Static"

PEBBLE View: "Dynamic"
Select a relevant performance metric, the Net Expected Benefit (NEB) (e.g. the net energy produced over a certain period).

Maximize NEB by:
intelligently shaping demand to perform generation-consumption matching subject to constraints (end-user thermal comfort, atypical availability of energy, reduced capacity demand at a certain time).

Applications:
EPB controller with 2 Switching Elements

Scenario	Linear Optimal controller vs Swicthing controller (% of improvement)
1	42,1%
2	67,3%
3	12,5%
4	37%

Problem 3: Practically Implementable LSS Design (LSS dynamics uncertain or even unknown)

- Combination of CAO and the Optimal-based (convex) approach
- At each time-step the optimal-based approach is employed to get an estimate of the system dynamics; based on the system dynamics estimate, the estimate of the Optimal-cost-to-go function is produced by employing the convex approach described previously.
- To make sure that the system dynamics estimate is an accurate one, the CAO approach is also employed at each time-step: many randomly-generated candidate control actions are generated and the one that «optimally» matches the HJB equation is chosen.
- E. Kosmatopoulos, *IEEE Transactions on Automatic Control*, 2008; E. Kosmatopoulos, *IEEE Transactions on Automatic Control*, provisionally accepted; E. Kosmatopoulos, *IEEE Transactions on Neural Networks*, provisionally accepted.

Problem 3: Practically Implementable LSS Design (LSS dynamics uncertain or even unknown)

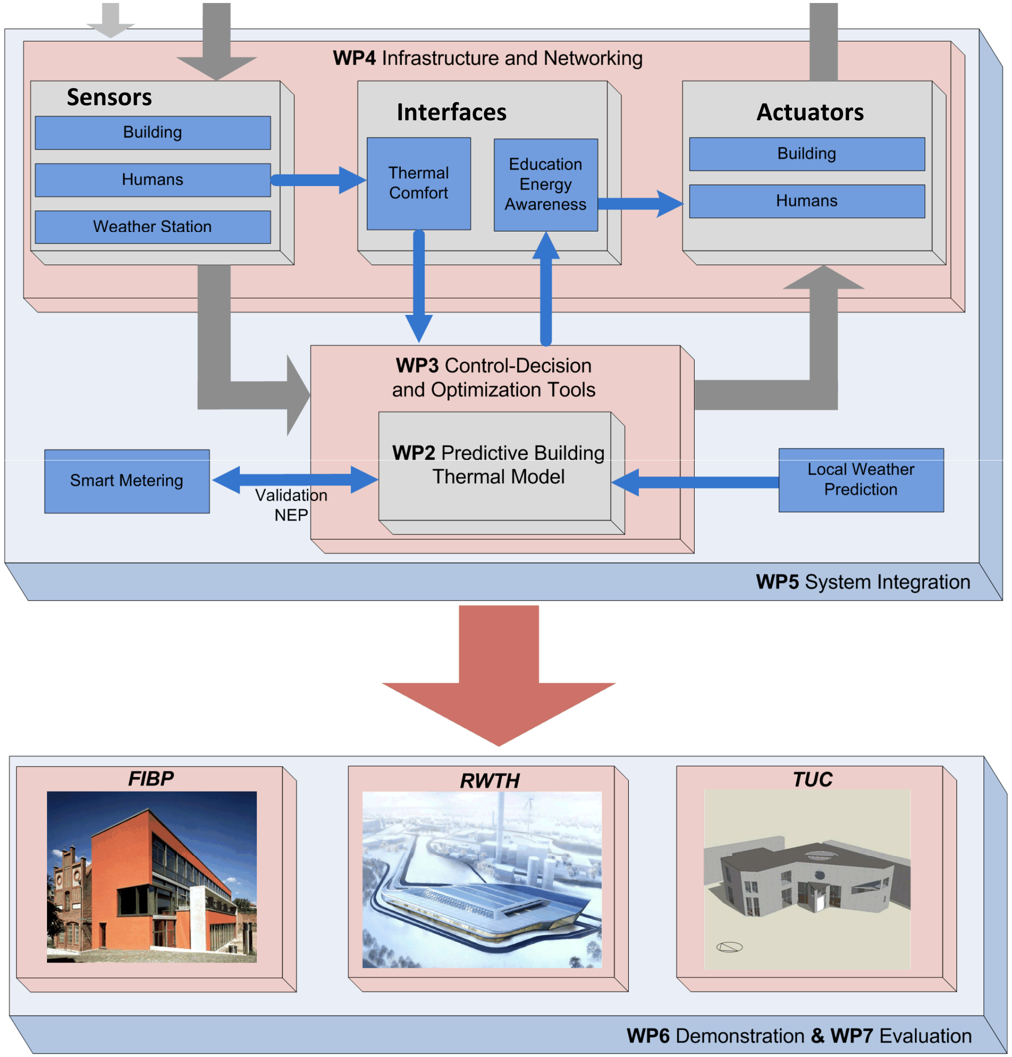
- Random generation of candidate controls guarantees persistence of excitation (crucial for learning)
- Arbitrarily close to the optimal performance
- Suitable for applications with re-configuration requirements
- Is it practicable?

The Challenge: Implementation and Evaluation in Real-Life Systems!

- FP7 sFLY (Swarms of Flying Robots)
- FP7 PEBBLE (Energy Positive Buildings)
- Traffic Control Systems (proposal under evaluation)

sFLY Project

PEBBLE Project:
…Conceptual Schematic



The PEBBLE approach to Building Optimization and Control