WP3 - Decentralized Control

SEVENTH FRAMEWORK PROGRAMME

European Commission
Information Society and Media

WIDE

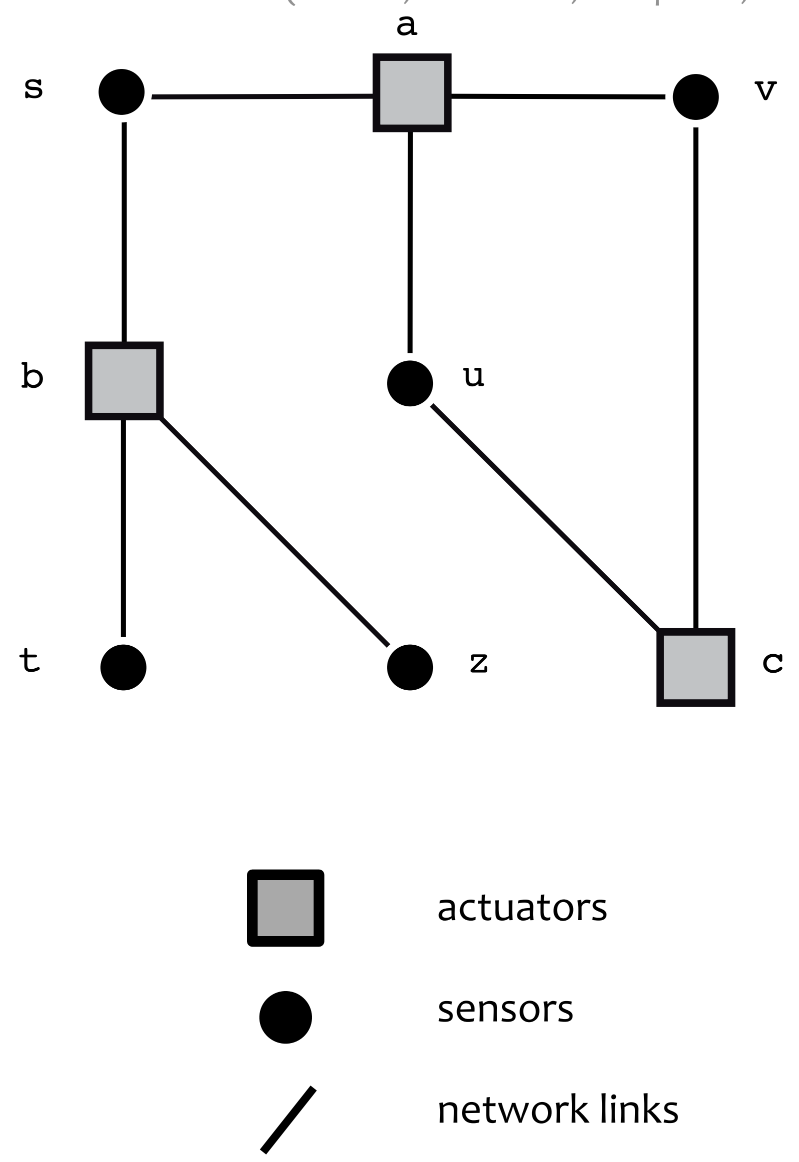
WP3 - Decentralized control activities (UNITN)

- **Decentralized MPC**: work presented during last telecon. SW included in WIDE Toolbox
- **Hierarchical + decentralized** control: approach now extended to enforce ***state constraints***. SW included in WIDE Toolbox (see example later in WIDE Toolbox presentation)
- **Decentralized linear** controller synthesis **via LMI** (SW included in WIDE Toolbox)
- Application to navigation of Unmanned Aerial Vehicles (UAVs)

Synthesis of decentralized linear controllers

(Barcelli, Bernardini, Bemporad, 2010)

- Spatially distributed system
  - linear model with coupled dynamics
  - sensors and actuators are distributed in space
  - distance limits transmission range
- Decentralized control setup
  - each controller commands a subset of the available actuators
  - each controller exploits a subset of the available sensors
  - states and inputs are subject to constraints



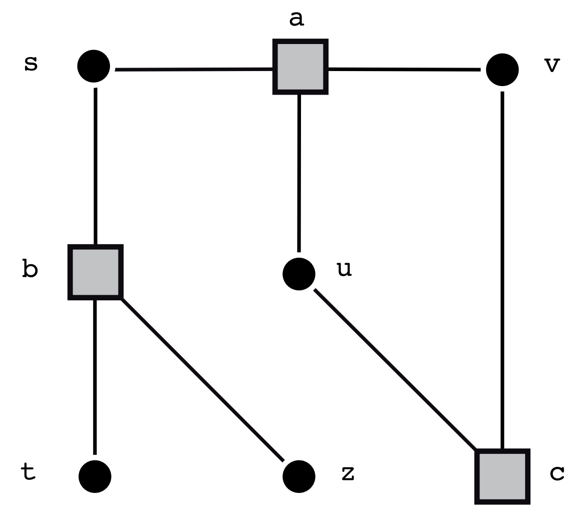
Decentralized framework

Plant model

$$x(t+1) \quad = \quad Ax(t) + Bu(t)$$

Constraints

$$\|x(t)\|_2 \le x_{\max}$$
$$\|u(t)\|_2 \le u_{\max}$$

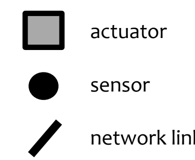


How to account for network topology?

adjacency matrix Λ

$$\lambda_{ij} = \begin{cases} 1 & \text{if sensor } s_j \text{ is linked to actuator } a_i, \\ 0 & \text{otherwise.} \end{cases}$$

$$\begin{array}{cc} & \begin{array}{ccccc} t & z & v & u & s \end{array} \\ \Lambda = & \left[\begin{array}{ccccc} 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{array}\right] \begin{array}{c} a \\ b \\ c \end{array} \end{array}$$



Linear decentralized regulator K

impose a structure on K

$$\lambda_{ij} = 0 \Rightarrow k_{ij} = 0,$$
$$i = 1, \ldots, m,$$
$$j = 1, \ldots, n$$

Decentralized linear controller synthesis

Theorem: Given network topology Λ, a *decentralized linear controller* $K = YQ^{-1}$ stabilizing the system under *state and input constraints* is obtained by solving the following ***semidefinite program***:

$$
\begin{aligned}
\min_{\gamma,Q,Y} \quad & \gamma \\
\text{s.t.} \quad & \begin{bmatrix} Q & * & * & * \\ AQ+BY & Q & * & * \\ Q_x^{1/2}Q & \mathbf{0} & \gamma I_n & * \\ Q_u^{1/2}Y & \mathbf{0} & \mathbf{0} & \gamma I_m \end{bmatrix} \succeq 0, \\
& \begin{bmatrix} Q & (AQ+BY)' \\ AQ+BY & x_{\max}^2 I_n \end{bmatrix} \succeq 0, \\
& \begin{bmatrix} u_{\max}^2 I_m & Y \\ Y' & Q \end{bmatrix} \succeq 0, \\
& \begin{bmatrix} 1 & v_i' \\ v_i & Q \end{bmatrix} \succeq 0, \; i = 1,\dots,n_v, \\
& \left.\begin{array}{l} (\lambda_{ij}=0) \Rightarrow y_{ij}=0 \\ (\lambda_{ij}=0)\wedge(\lambda_{ih}=1) \;\Rightarrow\; q_{hj}=0,\; q_{jh}=0 \end{array}\right\} \begin{array}{l} i=1,\dots,m, \\ j=1,\dots,n, \\ h=1,\dots,n \end{array}
\end{aligned}
$$

decreasing condition on Lyapunov function

state constraints satisfaction

input constraints satisfaction

initial condition $x(0) \in \mathcal{X}_0$ $\quad v = \text{vertex}(\mathcal{X}_0)$

decentralized structure

Explanation of main idea

$$\Lambda = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$ adjacency matrix

impose structure $Y = \begin{bmatrix} 0 & 0 & * & * & * \\ * & * & 0 & 0 & * \\ 0 & 0 & * & * & 0 \end{bmatrix}, \; Q = \begin{bmatrix} * & * & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 \\ 0 & 0 & * & * & 0 \\ 0 & 0 & * & * & 0 \\ 0 & 0 & 0 & 0 & * \end{bmatrix}$

$$K = YQ^{-1} = \begin{bmatrix} 0 & 0 & * & * & * \\ * & * & 0 & 0 & * \\ 0 & 0 & * & * & 0 \end{bmatrix}$$

Robust and stochastic extensions

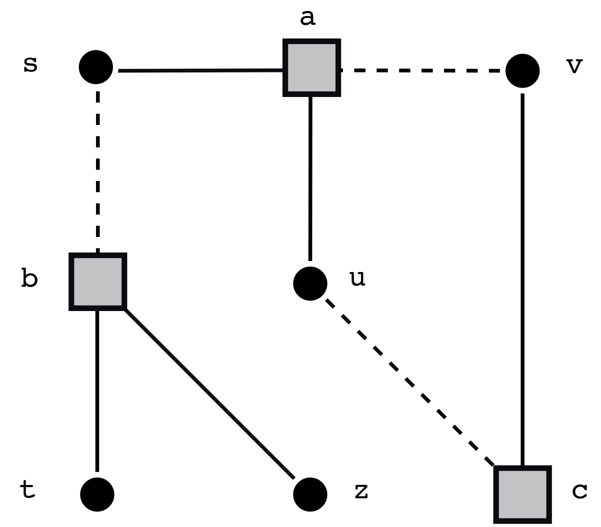
- Transmission over ***lossy*** links. Must consider a (combinatorial) number of possible scenarios.

$$\tilde{\Lambda}(t) \in \left\{\tilde{\Lambda}_1, \dots, \tilde{\Lambda}_\ell\right\}$$ network configuration @t

- Network configuration is only known ***locally***.

> **Idea:** design a decentralized controller that switches according to current **local** network configuration

$$u_i(t) = \begin{cases} F_1^i x(t) & \text{if } [\tilde{\Lambda}(t)]_i = \Gamma_1^i, \\ F_2^i x(t) & \text{if } [\tilde{\Lambda}(t)]_i = \Gamma_2^i, \\ \vdots & \vdots \\ F_{2^{l_i}}^i x(t) & \text{if } [\tilde{\Lambda}(t)]_i = \Gamma_{2^{l_i}}^i \end{cases}$$



network lossy links

$$\begin{array}{cc} & \begin{array}{ccccc} t & z & u & v & s \end{array} \\ \Lambda = & \begin{bmatrix} 0 & 0 & 1 & -1 & 1 \\ 1 & 1 & 0 & 0 & -1 \\ 0 & 0 & -1 & 1 & 0 \end{bmatrix} \begin{array}{c} a \\ b \\ c \end{array} \end{array}$$

- The set of controllers is designed simultaneously by solving an SDP problem

- Extension to **stochastic** setting, assuming 2-state Markov chain describes probability of packet loss

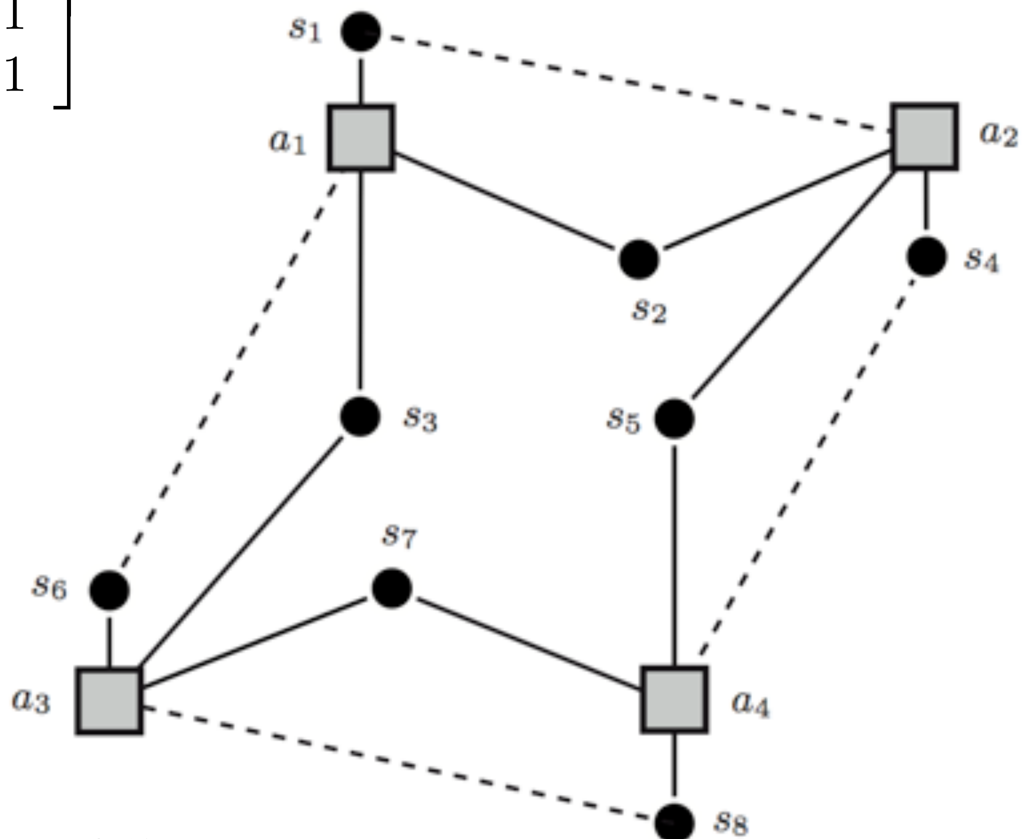
Simulation example: network topology

Randomly selected linear systems with fixed network topology

$$\Lambda = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Network possible configurations

$$\ell = 2^4 = 16$$



Simulation parameters

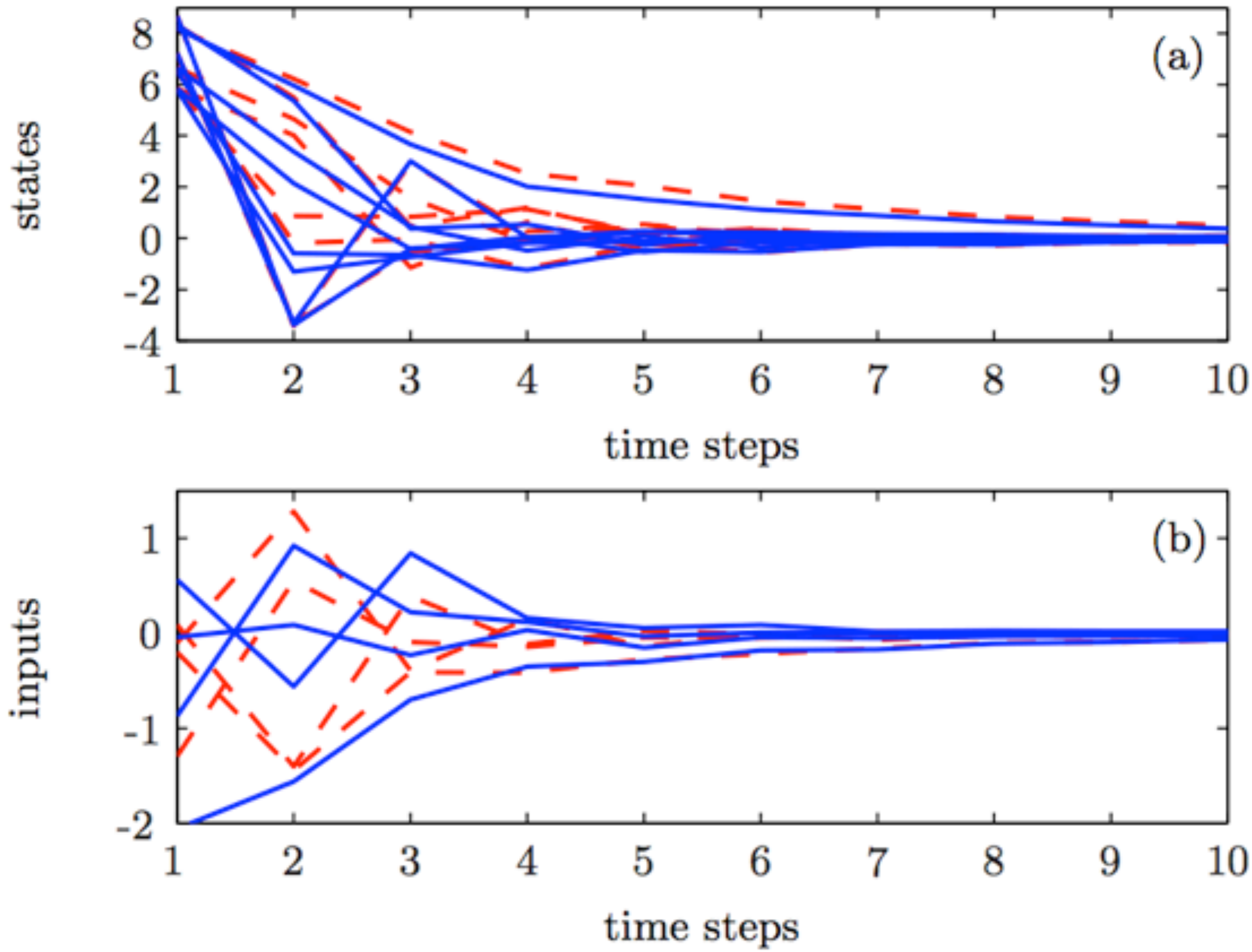
$$\begin{array}{ll} N_{sim} = 50, & T_{sim} = 50 \\ x_{max} = 25, & u_{max} = 3 \\ Q_x = I_n, & Q_u = 10^{-2} I_m \\ x(0) \in \mathcal{X}_0, & x_c = 7 \cdot \mathbf{1}_{8\times 1} \end{array}$$

$$\mathcal{X}_0 = \{x_c\} + \{x \in \mathbb{R}^8 : \ \|x\|_\infty \leq 2\}$$

Markov chain parameters

$$q_1 = 0.8,\ q_2 = 0.5,\ d_1 = 0.1 \text{ and } d_2 = 0.5$$

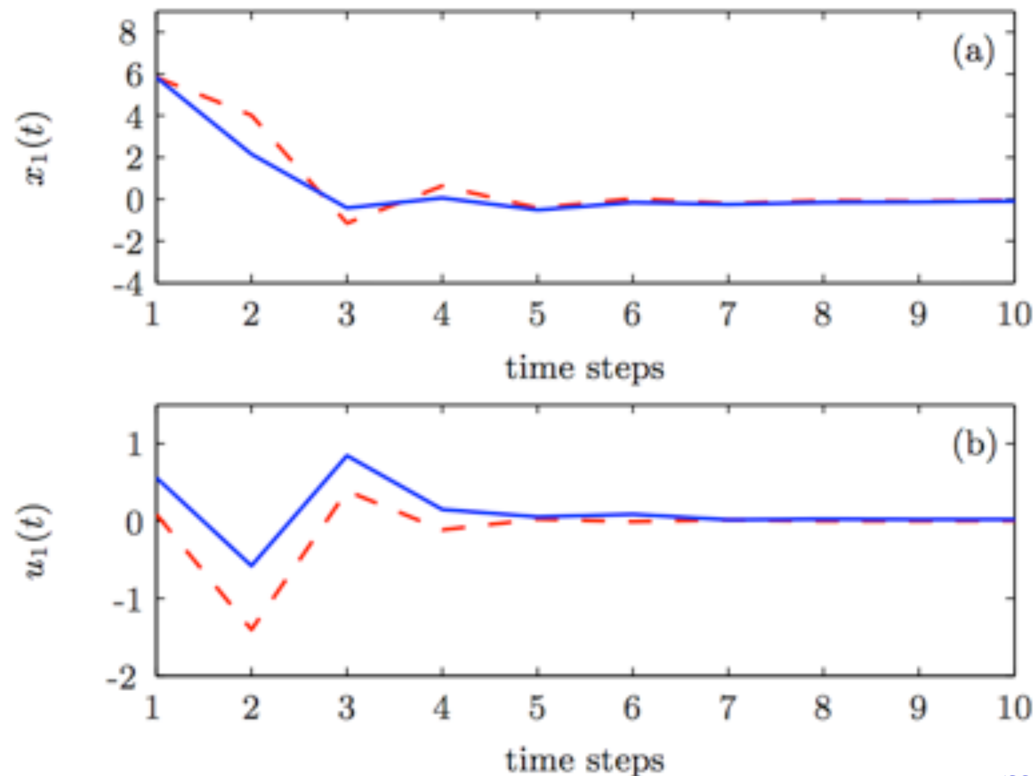
Simulation example: results



-- robust

— stochastic

Simulation example: results



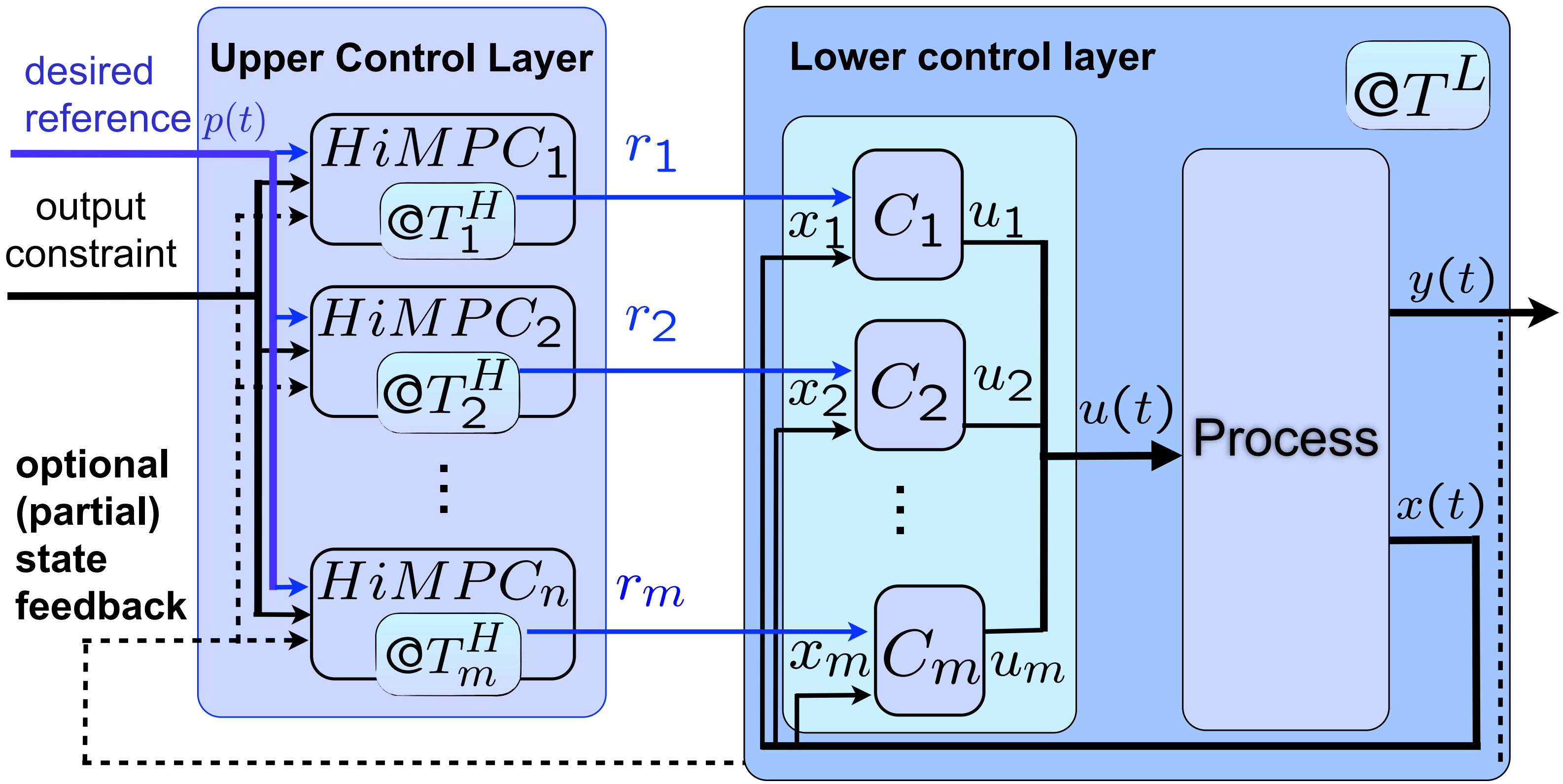
Cumulated storage cost

$$J_i = \sum_{t=1}^{T_{sim}} \left(\|Q_x x(t)\|_2 + \|Q_u u(t)\|_2 \right)$$

	mean $\mu(J_i)$	std. dev. $\sigma(J_i)$	CPU
Ideal network			(off-line time)
Centralized control	41.0	0	2.8 s
Decentralized control	45.1	0	1.2 s
Lossy network			(off-line time)
Dec. robust control	50.0	1.57	8.1 s
Dec. stochastic control	47.1	2.38	59.2 s

Decentralized Hierarchical MPC

(Barcelli, Bemporad, Ripaccioli, IFAC 2011)



- Two layers decentralized control structure, where each control loop is **independent** from the others and runs at individual sampling rate
- Lower control layer is in charge of stabilization
- State and reference constraints must be enforced

$$\mathcal{X}_i = \{ \left[\begin{smallmatrix} x_i \\ r_i \end{smallmatrix} \right] \in \mathbb{R}^{n_x^i + n_r^i} : \; H_x^i x_i + H_r^i r_i \leq K^i \}$$

Decentralized Hierarchical MPC

- **Main idea:** if all local controllers and reference signals are such that local states satisfy the constraints, then we can bound model uncertainty (=full model - decentralized model) within a polytope

$$\mathcal{D}_i = \{d_i \in \mathbb{R}^{n_x^i} : \exists x \in \mathbb{R}^{n_x},\ r \in \mathbb{R}^{n_r}$$
$$\text{such that}\ \ d_i = A_i^0 x + B_i^0 r,\ \left[\begin{smallmatrix} x \\ r \end{smallmatrix}\right] \in \mathcal{X}\}$$

- **Robust output admissible sets** are used then to generate reference signals

Decentralized Hierarchical MPC

The key idea is to follow the same steps as in HiMPC for each of the subsystem, exploiting the independence inherited from the MOARS.

Compute, for each subsystem, the minimal (in infinity norm) reference variation such that for any couple of references, initial state and sequence of disturbance realizations the state after N steps will lie outside the next MOARS.

For any smaller reference variation transitions among MOARS are guaranteed for each subsystem independently on the others.