Distributed LQR and Predictive Control

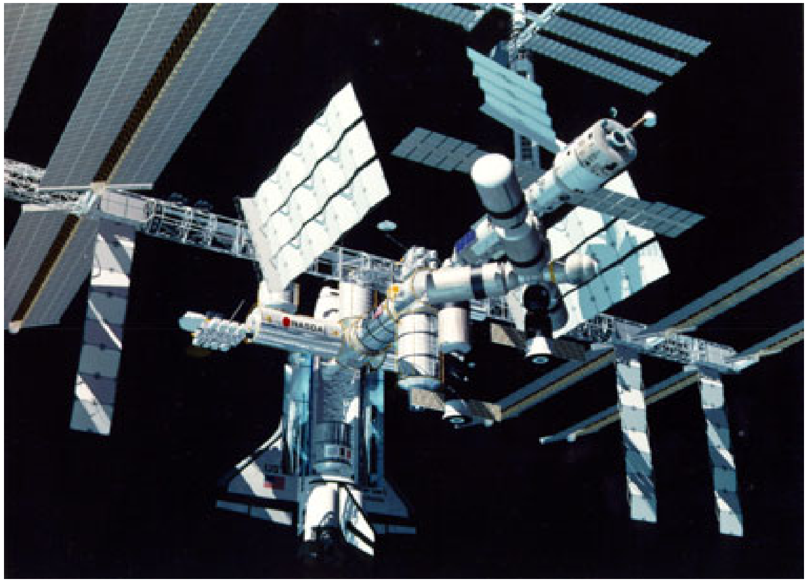
Tamás Keviczky

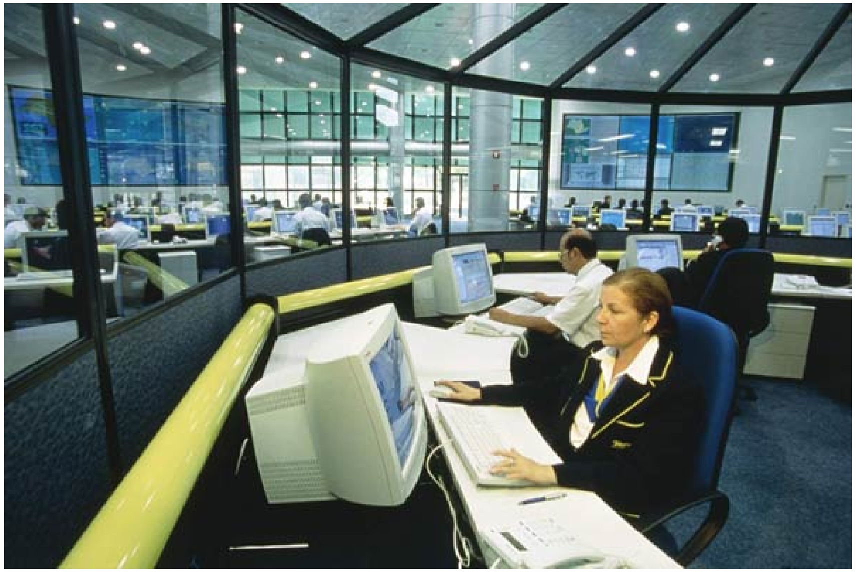
Postdoctoral Scholar
Control and Dynamical Systems
California Institute of Technology

Connections II

August 18, 2006

Today large-scale systems are ubiquitous …





Decentralized Control

- Long history
- Research today is motivated by
  - Technologies

    (micro-machinery, cheap wireless, fast comm. networks, …)
  - New computational tools

    (multiparametric solvers, invariant set calculations, …)
  - New methodologies

    (hybrid systems, explicit RHC, …)
- Problems have a structure

Common Features

- Large scale systems
- Dynamically decoupled
- Independently actuated
- Constrained subsystems

coupling from

Performance objective

+

Interconnection constraints

Common Control Objective

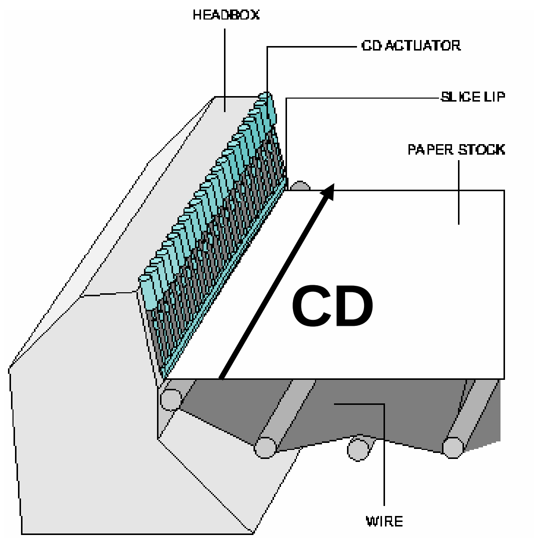
Decentralized control design which minimizes a certain performance index and satisfies interconnection constraints.

Focus on problems where *centralized solutions are prohibitive*

- too expensive
- not computationally feasible
- not feasible because of communication constraints

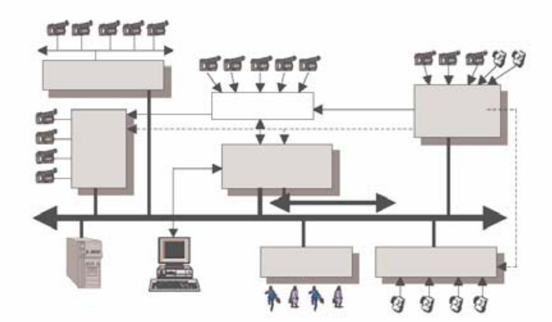
Motivating Examples

Cross-Directional Weight Profile Control



Organic Air Vehicle Formation Flight

Distributed PTZ Camera Systems



Problem Definition

- Collection of dynamical systems
  - decoupled
  - independently actuated

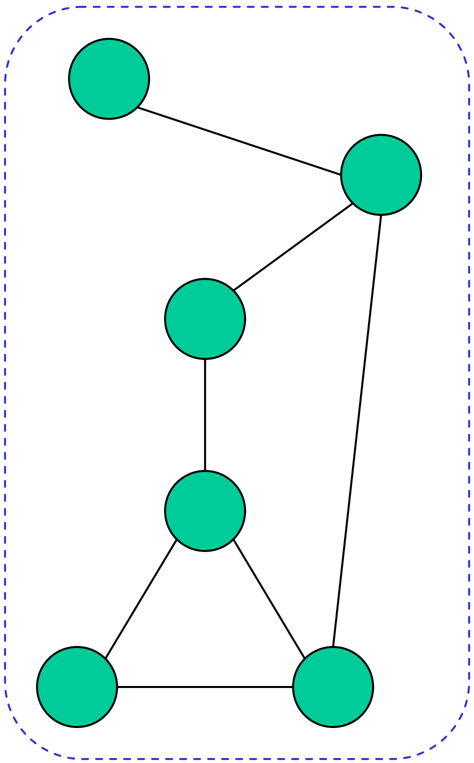
    $$x^i_{k+1} = f^i(x^i_k, u^i_k)$$

  - constrained

    $$x^i_k \in \mathcal{X}^i,\ \ u^i_k \in \mathcal{U}^i$$

- Optimal control problem
  - coupling in performance objective
  - coupling in constraints
- Graph structure for describing:
  - information exchange
  - constraints between nodes
  - coupling in performance objective

$$\mathcal{G}_k = \{\mathcal{V}_k, \mathcal{A}_k\}$$



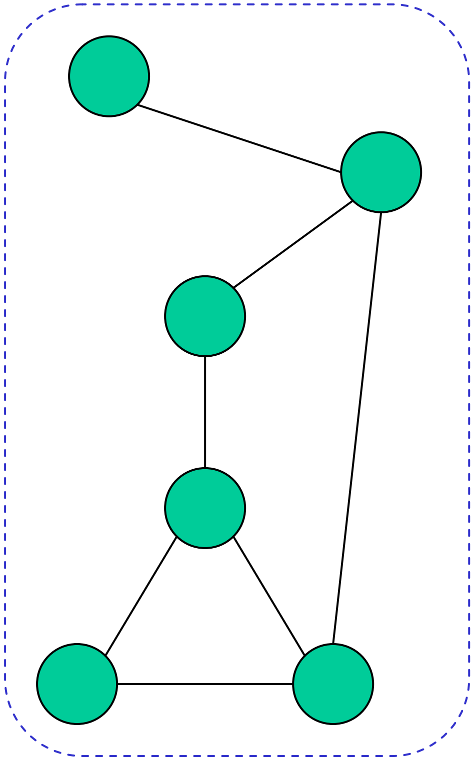
Problem Definition

Optimal control problem

$$\min_{\{\tilde{u}_0,\tilde{u}_1,\ldots\}} \sum_{k=0}^{\infty} l(\tilde{x}_k, \tilde{u}_k)$$

$$\text{subj. to} \begin{cases} x^i_{k+1} = f^i(x^i_k, u^i_k), \\ g^{i,j}(x^i_k, u^i_k, x^j_k, u^j_k) \leq 0, \\ \quad i = 1, \ldots, N_v, \ (i,j) \in \mathcal{A}_k \ \ k \geq 0 \\ x^i_k \in \mathcal{X}^i, \ \ u^i_k \in \mathcal{U}^i, \ k \geq 0 \end{cases}$$

$\mathcal{G}_k = \{\mathcal{V}_k, \mathcal{A}_k\}$



Coupling in performance objective

$$l(\tilde{x}, \tilde{u}) = \sum_{i=1}^{N_v} l^i(x^i, u^i, \tilde{x}^i, \tilde{u}^i)$$

Coupling in constraints

$$g^{i,j}(x^i, u^i, x^j, u^j) \leq 0, \quad (i,j) \in \mathcal{A}$$

Common Features

- Large scale systems
- Dynamically decoupled
- Independently actuated
- ~~Constrained subsystems~~
- Identical LTI subsystems

coupling from

Quadratic Performance objective

+

~~Interconnection constraints~~

Distributed LQR Design

- Identical unconstrained linear models

$$\dot{x}_i = Ax_i + Bu_i, \qquad i = 1, \ldots, N$$

- Minimization of absolute and relative state errors

$$J = \int_{t=0}^{\infty} \left(\sum_{i=1}^{N} x_i(t)^T Q_a x_i(t) + u_i(t)^T R u_i(t) + \sum_{i=1}^{N} \sum_{j|(i,j)\in\mathcal{A}} \left(x_i(t) - x_j(t)\right)^T Q_{ij} \left(x_i(t) - x_j(t)\right) \right) dt$$

- Global LQR problem

$$\tilde{A} = I_N \otimes A$$

$$\tilde{B} = I_N \otimes B$$

$$\tilde{R} = I_N \otimes R$$

$$\tilde{Q} = \left[\tilde{Q}_{ij}\right] \begin{cases} \tilde{Q}_{ii} = Q_a + \sum_{k|(i,k)\in\mathcal{A}} Q_{ik}, & \forall i = 1, \ldots, N \\ \tilde{Q}_{ij} = -Q_{ij}, & \forall (i,j) \in \mathcal{A} \end{cases}$$

Sparse structure, corresponding to an arbitrary undirected communication graph $\mathcal{G}$

Find stabilizing linear controller,
which minimizes J and has same structure of $\tilde{Q}$

Distributed LQR Design

- We can easily construct a *stabilizing* distributed controller with same structure of the Laplacian.

- This is independent from choice of Q_a, Q_{ij} and R, which can be used for tuning.

- Can be obtained by solving one local, fully connected LQR subproblem with dimension $N_l = \max(d_i) + 1$.
  - Based on special properties of the local ARE solution $\tilde{K} = \left[\tilde{K}_{ij}\right]$, $\tilde{P} = \left[\tilde{P}_{ij}\right]$

1.) $\sum_{j=1}^{N_l} \tilde{K}_{ij} = K, \quad \sum_{j=1}^{N_l} \tilde{P}_{ij} = P, \quad \forall i = 1, \ldots, N_l,$

where K, P solve $\mathrm{LQR}(A, B, Q_a, R)$.

2.) $\tilde{P}$ is a block M-matrix.

[Borrelli – Keviczky, CDC'06]

Distributed Stabilizing Control from Local LQR Solution

- Consider original cost function with two types of weights: Q_a , Q_r

$$J = \int_{t=0}^{\infty} \left(\sum_{i=1}^{N} x_i(t)^T Q_a x_i(t) + u_i(t)^T R u_i(t) + \sum_{i=1}^{N} \sum_{j|(i,j)\in\mathcal{A}} \left(x_i(t) - x_j(t)\right)^T Q_r \left(x_i(t) - x_j(t)\right) \right) dt$$

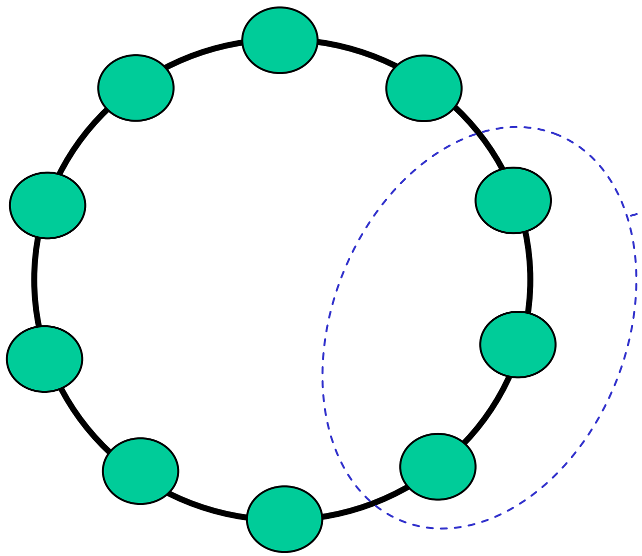
- Solve small, local, fully connected problem with

$$J = \int_{t=0}^{\infty} \left(\sum_{i=1}^{d_{\max}+1} x_i(t)^T Q_a x_i(t) + u_i(t)^T R u_i(t) + \sum_{i=1}^{d_{\max}+1} \sum_{j\neq i}^{d_{\max}+1} \left(x_i(t) - x_j(t)\right)^T Q_r \left(x_i(t) - x_j(t)\right) \right) dt$$

$$\tilde{Q} = \begin{bmatrix} \tilde{Q}_1 & \tilde{Q}_2 & \cdots & \cdots & \tilde{Q}_2 \\ \tilde{Q}_2 & \tilde{Q}_1 & \ddots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \ddots & \tilde{Q}_1 & \tilde{Q}_2 \\ \tilde{Q}_2 & \cdots & \cdots & \tilde{Q}_2 & \tilde{Q}_1 \end{bmatrix} \Rightarrow \tilde{K} = \begin{bmatrix} \tilde{K}_1 & \tilde{K}_2 & \cdots & \cdots & \tilde{K}_2 \\ \tilde{K}_2 & \tilde{K}_1 & \ddots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \ddots & \tilde{K}_1 & \tilde{K}_2 \\ \tilde{K}_2 & \cdots & \cdots & \tilde{K}_2 & \tilde{K}_1 \end{bmatrix}$$

- Construct stabilizing controller as $\hat{K} = I_N \otimes \tilde{K}_1 - \mathbf{A} \otimes \tilde{K}_2$

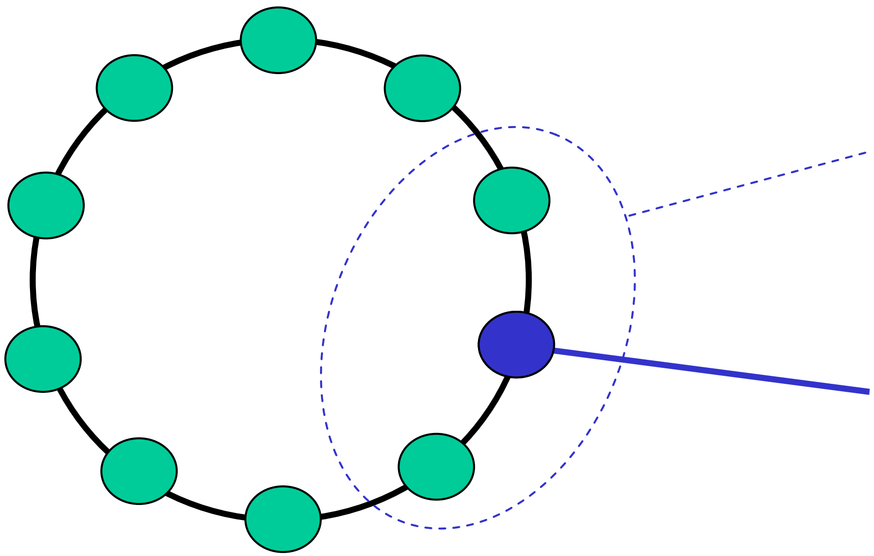
Distributed Stabilizing Control from Local LQR Solution



$$u(t)=\begin{bmatrix}\tilde{K}_1 & \tilde{K}_2 & \tilde{K}_2\\ \tilde{K}_2 & \tilde{K}_1 & \tilde{K}_2\\ \tilde{K}_2 & \tilde{K}_2 & \tilde{K}_1\end{bmatrix}x(t)$$

$$\tilde{Q}=\begin{bmatrix}Q_a+2Q_r & -Q_r & -Q_r\\ -Q_r & Q_a+2Q_r & -Q_r\\ -Q_r & -Q_r & Q_a+2Q_r\end{bmatrix}$$

Distributed Stabilizing Control from Local LQR Solution



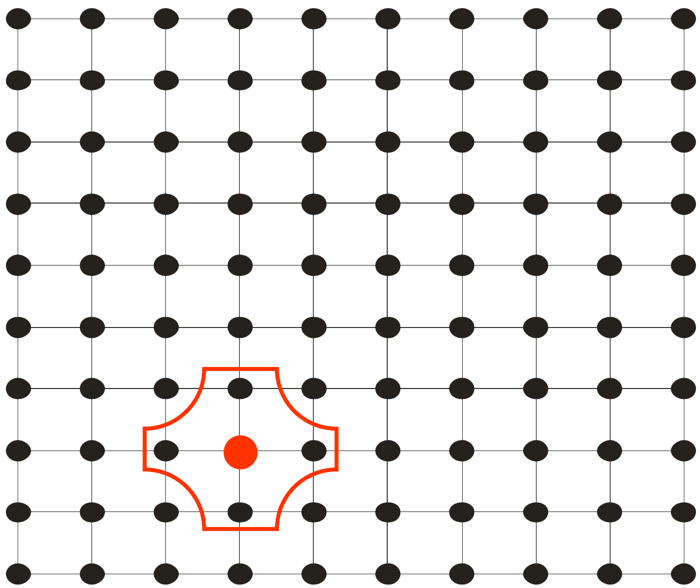
$$u(t)=\begin{bmatrix}\cancel{\tilde{K}_1} & \cancel{\tilde{K}_2} & \cancel{\tilde{K}_2}\\ \tilde{K}_2 & \tilde{K}_1 & \tilde{K}_2\\ \cancel{\tilde{K}_2} & \cancel{\tilde{K}_2} & \cancel{\tilde{K}_1}\end{bmatrix}x(t)$$

$$K_{\text{local}}=\begin{bmatrix}\tilde{K}_2 & \tilde{K}_1 & \tilde{K}_2\end{bmatrix}$$

Stabilizing distributed controller: $\hat{K}=I_N\otimes\tilde{K}_1-\mathbf{A}\otimes\tilde{K}_2$

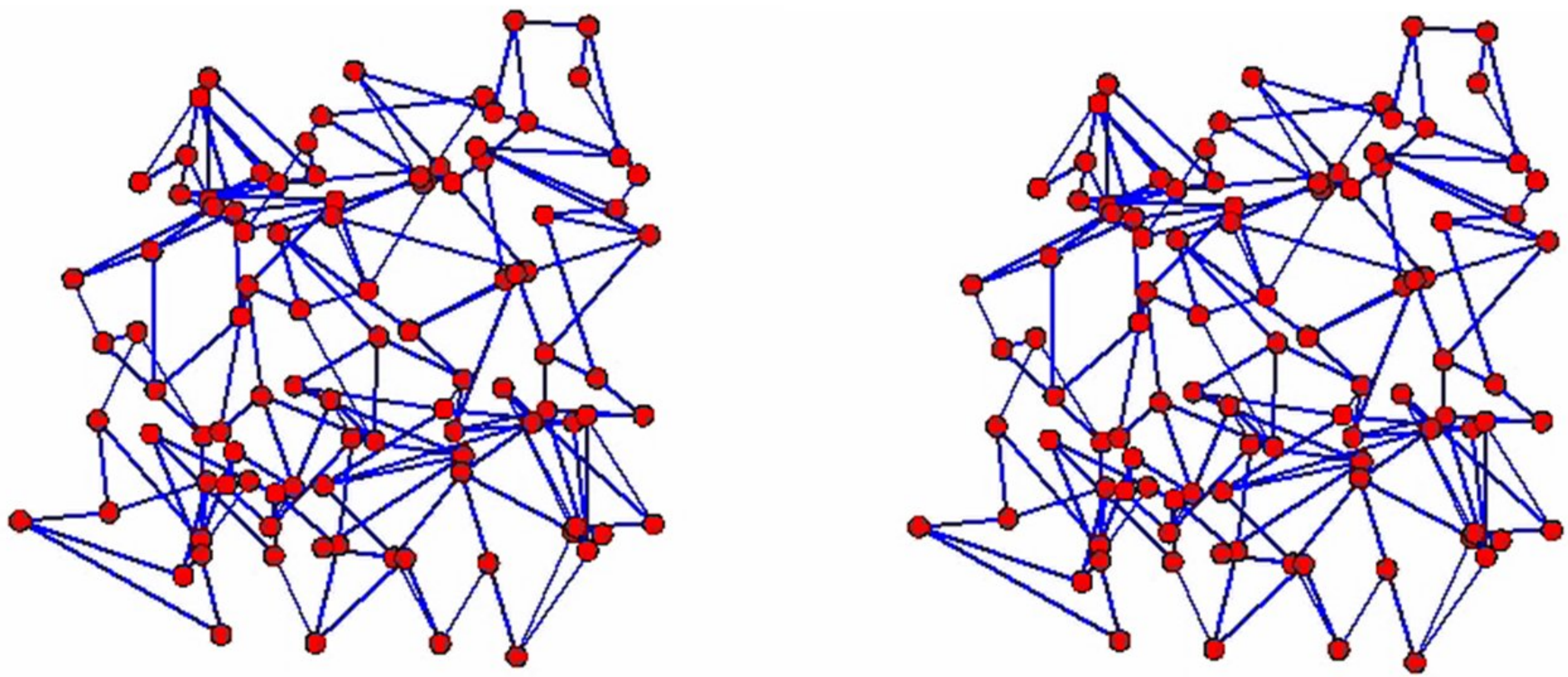
$$\hat{K}=\begin{bmatrix}\tilde{K}_1 & \tilde{K}_2 & 0 & \cdots & \tilde{K}_2\\ \tilde{K}_2 & \tilde{K}_1 & \tilde{K}_2 & \cdots & 0\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & \cdots & \tilde{K}_2 & \tilde{K}_1 & \tilde{K}_2\\ \tilde{K}_2 & \cdots & 0 & \tilde{K}_2 & \tilde{K}_1\end{bmatrix}\qquad \mathbf{L}=\begin{bmatrix}2 & -1 & 0 & \cdots & -1\\ -1 & 2 & -1 & \cdots & 0\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & \cdots & -1 & 2 & -1\\ -1 & \cdots & 0 & -1 & 2\end{bmatrix}$$

Example with Simple Systems on a Lattice

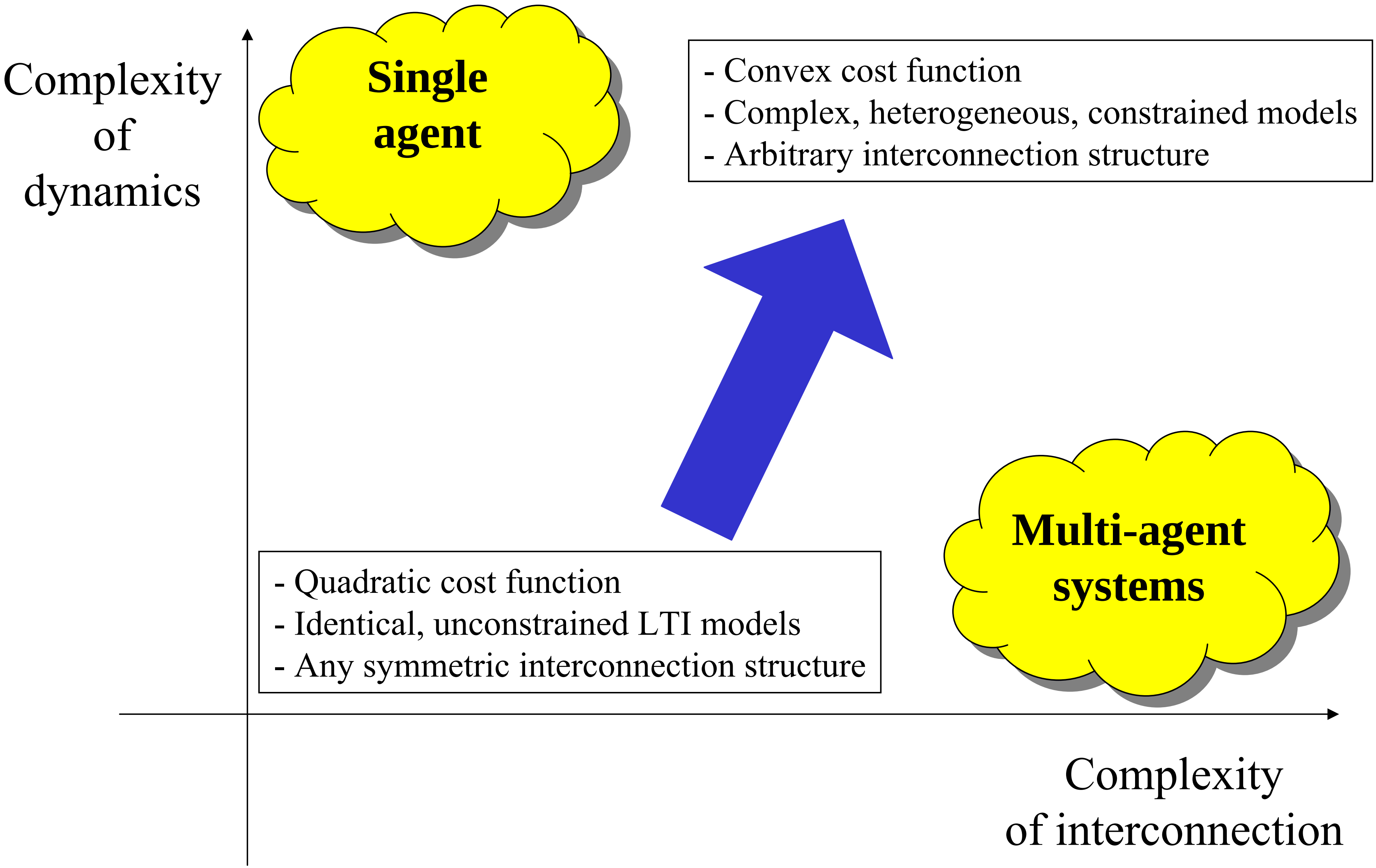


- Construct stabilizing distributed controller (absolute and relative state references)
- Use solution of one local LQR problem involving only five subsystems ($d_{\max} + 1$)
- Show effect of different Q_a , Q_r choices

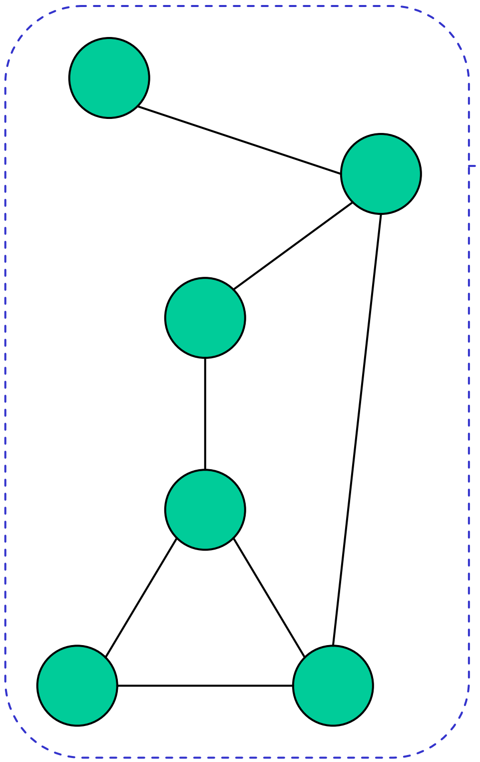
Example with Simple Systems on a Lattice



Increasing Complexity



Infinite Time, Centralized Problem

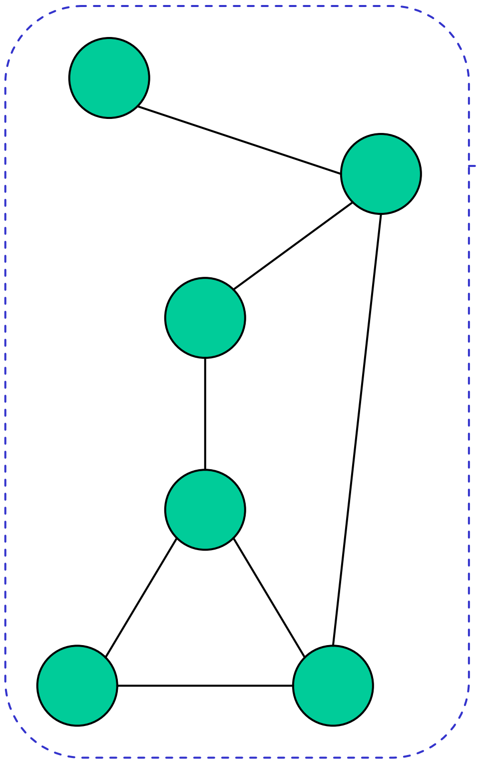


$$\min_{\{\tilde{u}_0,\tilde{u}_1,\ldots\}} \sum_{k=0}^{\infty}$$ **Global performance index**

subj. to

- **Models of subsystems**
- **Physical constraints of subsystems**
- **Interaction constraints**

Finite Time, Centralized RHC Problem



$$\min_{\{U_t\}} \sum_{k=0}^{N-1} \textbf{Global perf. index and terminal cost}$$

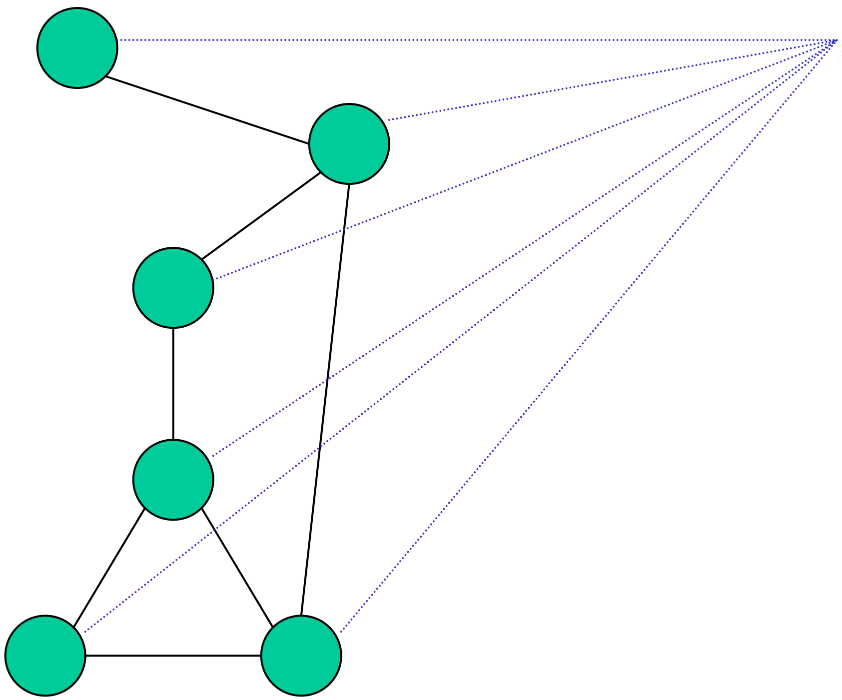
subj. to

- **Models of subsystem**
- **Physical co… of subsystems**
- **Int… constraints**
- **…rminal constraints**

Computationally prohibitive

Simple idea:
- – Break centralized RHC controller into local problems of smaller size.
- – Use information about neighbors, predict neighbors' trajectories. (Driving in traffic analogy.)
- – Implement own control solution.

Decentralized RHC Scheme



$$
\min_{\tilde{U}_t^i} \sum_{k=0}^{N-1} \overbrace{l^i(x^i_{k,t}, u^i_{k,t}, \tilde{x}^i_{k,t}, \tilde{u}^i_{k,t}) + l^i_N(x^i_{N,t}, \tilde{x}^i_{N,t})}^{\text{Local performance index}}
$$

$$
\text{subj. to} \left\{
\begin{array}{ll}
\left.\begin{array}{l}
x^i_{k+1,t} = f^i(x^i_{k,t}, u^i_{k,t}), \;\; k \geq 0 \\
x^i_{k,t} \in \mathcal{X}^i, \quad u^i_{k,t} \in \mathcal{U}^i, \\
\qquad k = 1, \ldots, N-1
\end{array}\right\} & \text{Own model and constraints} \\
\left.\begin{array}{l}
x^j_{k+1,t} = f^j(x^j_{k,t}, u^j_{k,t}), \\
\qquad (j,i) \in \mathcal{A}, \;\; k \geq 0 \\
x^j_{k,t} \in \mathcal{X}^j, \quad u^j_{k,t} \in \mathcal{U}^j, \\
\qquad (j,i) \in \mathcal{A}, \\
\qquad k = 1, \ldots, N-1
\end{array}\right\} & \text{Neighbors' model and constraints} \\
\left.\begin{array}{l}
g^{i,j}(x^i_{k,t}, u^i_{k,t}, x^j_{k,t}, u^j_{k,t}) \leq 0, \\
\qquad (i,j) \in \mathcal{A}, \\
\qquad k = 1, \ldots, N-1
\end{array}\right\} & \text{Interaction constr. with neighbors,} \\
\left.\begin{array}{l}
g^{q,r}(x^q_{k,t}, u^q_{k,t}, x^r_{k,t}, u^r_{k,t}) \leq 0, \\
\qquad (q,i) \in \mathcal{A}, \;\; (r,i) \in \mathcal{A}, \\
\qquad k = 1, \ldots, N-1
\end{array}\right\} & \text{between neighbors} \\
x^i_{N,t} \in \mathcal{X}^i_f, \quad x^j_{N,t} \in \mathcal{X}^j_f, (i,j) \in \mathcal{A} & \\
x^i_{0,t} = x^i_t, \quad \tilde{x}^i_{0,t} = \tilde{x}^i_t, &
\end{array}
\right.
$$

Advantages

- Flexibility
  - Different objectives in cost
    (e.g. maneuvering, formation keeping, joining, flying etc.)

- Explicit incorporation of constraints
  - Allows systematical study of feasibility
    (e.g. collision avoidance)

- Real-time implementation
  - For certain classes of systems and constraints
    (e.g. using equivalent PWA controller)

but

The problem formulation by itself does not guarantee stability and feasibility !!!

Main Issues in Decentralization of RHC

- Neighbors’ predictions can be wrong
  - Even if the idea of predicting the neighbors’ behaviors is intuitive and can be observed in practice (e.g. driving in traffic, birds, etc.)
- Impossible to avoid conservativeness
  - How conservative?
  - Worst-case scenario not applicable for such class of problems.
  - Information exchange, cooperation becomes important.

Stability Analysis

- Approach
  - Use value function of individual nodes (subsystems) as Lyapunov functions.
  - Has long been used for stability of interconnected systems with certain bounds on the interactions.
- Prediction mismatch drives the problem
  - Multiple optima
    (non-strictly convex cost function, non-convex constraints)
  - Graph structure
    (different set of neighbors)

prediction mismatch $\longrightarrow \displaystyle\sum_{j \mid (i,j) \in \mathcal{A}} \varepsilon^{i,j} \leq J_0^{i*} \longleftarrow$ initial conditions

stability condition

Stability Analysis

$$\sum_{j \mid (i,j) \in \mathcal{A}} \varepsilon^{i,j} \leq J_0^{i*}$$

Initial conditions

$$J_0^{i*} = \|Qx_0^i\|_p + \|Ru_{0,0}^{i,i}\|_p + \sum_{j \mid (i,j) \in \mathcal{A}} \|Qx_0^j\|_p + \sum_{j \mid (i,j) \in \mathcal{A}} \|Ru_{0,0}^{j,i}\|_p$$
$$+ \sum_{j \mid (i,j) \in \mathcal{A}} \|Q(x_0^i - x_0^j)\|_p$$

Stability Analysis

$$\sum_{j \mid (i,\, j) \in \mathcal{A}} \varepsilon^{i,\, j} \leq J_0^{i*}$$

Prediction mismatch

$$\varepsilon^{i,j} = \sum_{k=1}^{N-1} \left(2\|Q(x_{k,0}^{j,j} - x_{k,0}^{j,i})\|_p + \|R(u_{k,0}^{j,j} - u_{k,0}^{j,i})\|_p \right)$$

Stability Analysis

$$\underbrace{\sum_{j \mid (i,\, j) \in \mathcal{A}} \varepsilon^{i,\, j} \leq J_0^{i*}}$$

$\Big\downarrow$ + terminal point constraint

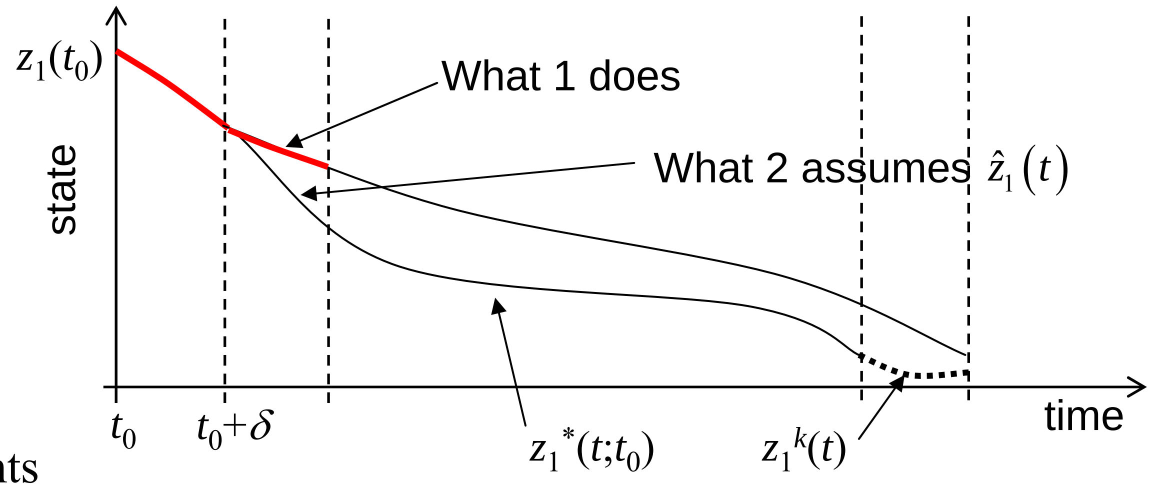
Asymptotic stability of each node

Testing the condition

- Unconstrained LTI: checking $M_i \geq 0$ (M_i is of limited size)
- Constrained LTI: solve systems of LMIs (limited size)

Exchange of Optimizers for Stability

[Dunbar - Murray]



- **Subsystems**
  - Independently actuated, dynamically decoupled
  - Independently constrained
- **Coupling**
  - Performance objective
  - NO interconnection constraints
- **Stability can be obtained using**
  - A given cost structure
  - Sufficiently fast, synchronous updates
  - Exchange of most recent optimal control trajectory between coupled subsystems
  - Compatibility constraint: stay within bounded path of what was transmitted

$$\min_{u_1(\cdot)} \int_{t_k}^{t_k+T} L_1\left(z_1(\tau), \hat{z}_2(\tau), u_1(\tau)\right) d\tau + G_1\left(z_1(t_k+T)\right)$$

$$\text{s.t.} \quad \dot{z}_1(t) = f_1\left(z_1(t), u_1(t)\right)$$

$$u_1(t) \in U_1, \quad z_1(t_k+T) \in Z_{f_1}$$

$$\left\| z_1(t) - \hat{z}_1(t) \right\| \le \delta^2 \kappa$$

Approaches to Address Feasibility of Coupling Constraints

- **Communication**
  - Exchange of optimizers between neighbors

  [Dunbar – Murray]
  [Richards – How]

- **Robust constraint fulfillment**
  - Decoupled terminal regions don't work
    (perhaps stabilizing decentralized LTI terminal controllers)
  - Worst-case approaches (too conservative)
  - Time-varying, increasing uncertainty about neighbors

  [Jia, Krogh]

- **Implicit safety guarantees**
  - Feasible basis states as hard terminal constraints in receding horizon path planning

  [Schouwenaars]

- **Hierarchy in the interconnection graph** $\mathcal{G}$
  - Feasible set projection

  [Gokbayrak – Cassandras]
  [Stipanovic – Tomlin]
  [Keviczky – Borrelli – Balas, CDC'04]

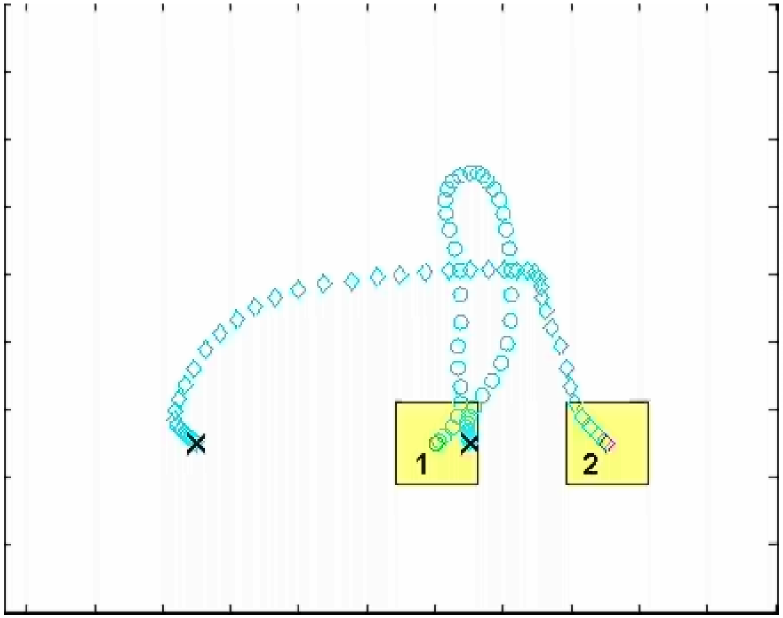
- **Recovering from infeasibility**
  - Emergency maneuvers using invariant sets

  [Borrelli – Keviczky – Balas, CDC'04]

- **Hybrid receding horizon control**
  - Include "right-of-way" coordination rules
    in problem formulation using binary decision variables

  [Keviczky – Vanek – Borrelli – Balas, HSCC'05, ACC'06]

2-Vehicle Formation

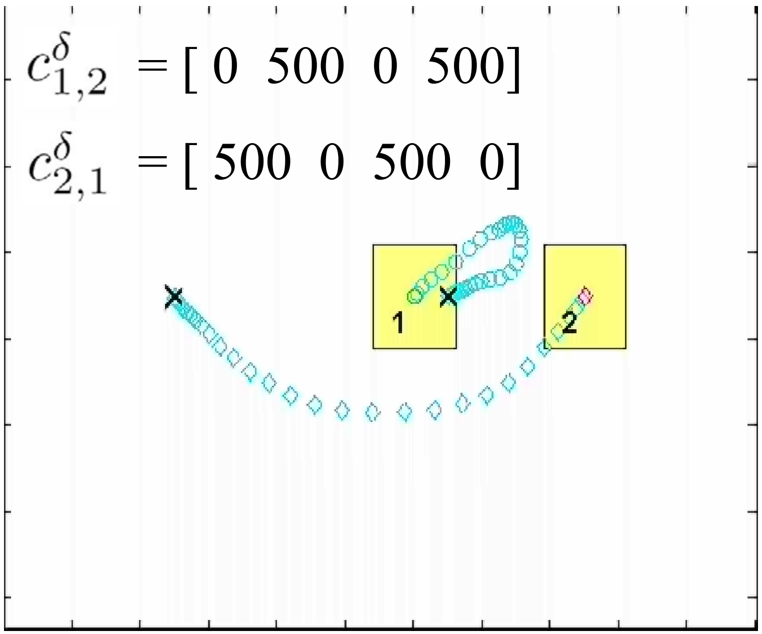


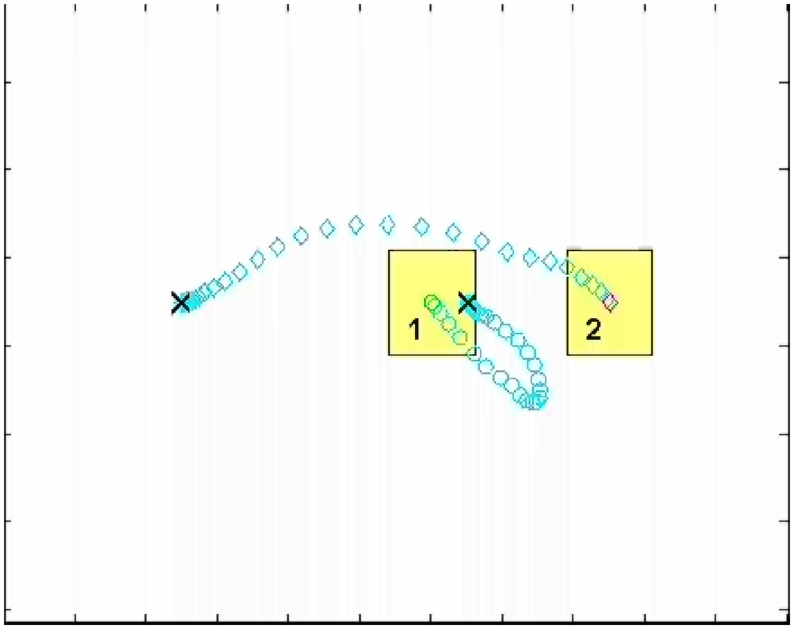
$$J_N^{bin}(x_t^i, \tilde{x}_t^i) \triangleq \sum_{k=0}^{N-1} \left(l_t^i(x_{k,t}^i, u_{k,t}^i, \tilde{x}_{k,t}^i, \tilde{u}_{k,t}^i) + l_N^i(x_{N,t}^i, \tilde{x}_{N,t}^i) \right) + \sum_{k=0}^{N-1} \sum_{j|(i,j)\in\mathcal{A}(t)} c_{i,j}^{\delta} \delta_{i,j}(k)$$

$$\delta_{i,j} = [\delta_{i,j}^E \quad \delta_{i,j}^W \quad \delta_{i,j}^N \quad \delta_{i,j}^S]'$$

$$c_{1,2}^{\delta} = [\,0 \;\; 500 \;\; 0 \;\; 500]$$

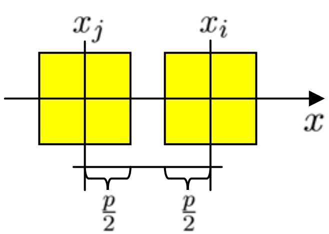
$$c_{2,1}^{\delta} = [\,500 \;\; 0 \;\; 500 \;\; 0]$$

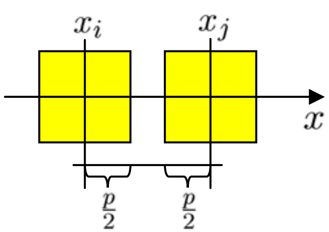




Coordination Through Rules

- Protection zones modeled as parallelepipeds [Borrelli, Keviczky, Balas et al. in CDC'04, HSCC'05]
- Disjunctions represented by binary decision variables [Schouwenaars ECC'01]
  - Describe the location of a vehicle with respect to the protection zone of another.





Disjunction inequality	Binary variable	Big-M technique
$1: \quad x_i - \frac{p}{2} \geq x_j + \frac{p}{2}$	$\delta_{ij}^E = \begin{cases} 1 \text{ if ineq. TRUE} \\ 0 \text{ if ineq. FALSE} \end{cases}$	$x_j - x_i + p \leq M(1 - \delta_{ij}^E)$ $x_j - x_i + p > m\delta_{ij}^E$
$2: \quad x_i + \frac{p}{2} \leq x_j - \frac{p}{2}$	$\delta_{ij}^W = \begin{cases} 1 \text{ if ineq. TRUE} \\ 0 \text{ if ineq. FALSE} \end{cases}$	$x_i - x_j + p \leq M(1 - \delta_{ij}^W)$ $x_i - x_j + p > m\delta_{ij}^W$
$3: \quad y_i - \frac{p}{2} \geq y_j + \frac{p}{2}$	$\delta_{ij}^N = \begin{cases} 1 \text{ if ineq. TRUE} \\ 0 \text{ if ineq. FALSE} \end{cases}$	$y_j - y_i + p \leq M(1 - \delta_{ij}^N)$ $y_j - y_i + p > m\delta_{ij}^N$
$4: \quad y_i + \frac{p}{2} \leq y_j - \frac{p}{2}$	$\delta_{ij}^S = \begin{cases} 1 \text{ if ineq. TRUE} \\ 0 \text{ if ineq. FALSE} \end{cases}$	$y_i - y_j + p \leq M(1 - \delta_{ij}^S)$ $y_i - y_j + p > m\delta_{ij}^S$

For collision avoidance: $\delta_{ij}^E \ \texttt{OR} \ \delta_{ij}^W \ \texttt{OR} \ \delta_{ij}^N \ \texttt{OR} \ \delta_{ij}^S$

Decentralized Hybrid RHC

Rule element:

Boolean-valued function of states of a node and its neighbors' states

$$\varrho \;:\; (x^i, \tilde{x}^i) \rightarrow X, \qquad X = \{true, false\}\,.$$

Rule:

Propositional logic statement involving rule elements

$$\mathcal{R} \;:\; (\varrho_1, \varrho_2, \ldots, \varrho_{n-1}) \rightarrow X, \qquad X = \{true, false\}\,.$$

The rule is not respected and its value is “false” when the underlying statement is false.

Coordinating functions:

Operate on a set of rules and the states of a node and its neighbors

$$\mathscr{F}_c^C \;:\; (\Re, x^i, \tilde{x}^i) \rightarrow \mathbb{R}$$

$$\mathscr{F}_c^{bin} \;:\; (\Re, x^i, \tilde{x}^i) \rightarrow \{0, 1\}$$

Coordinating Functions

- **Can have either continuous or binary values**
  They can be included in the cost function or constraints.

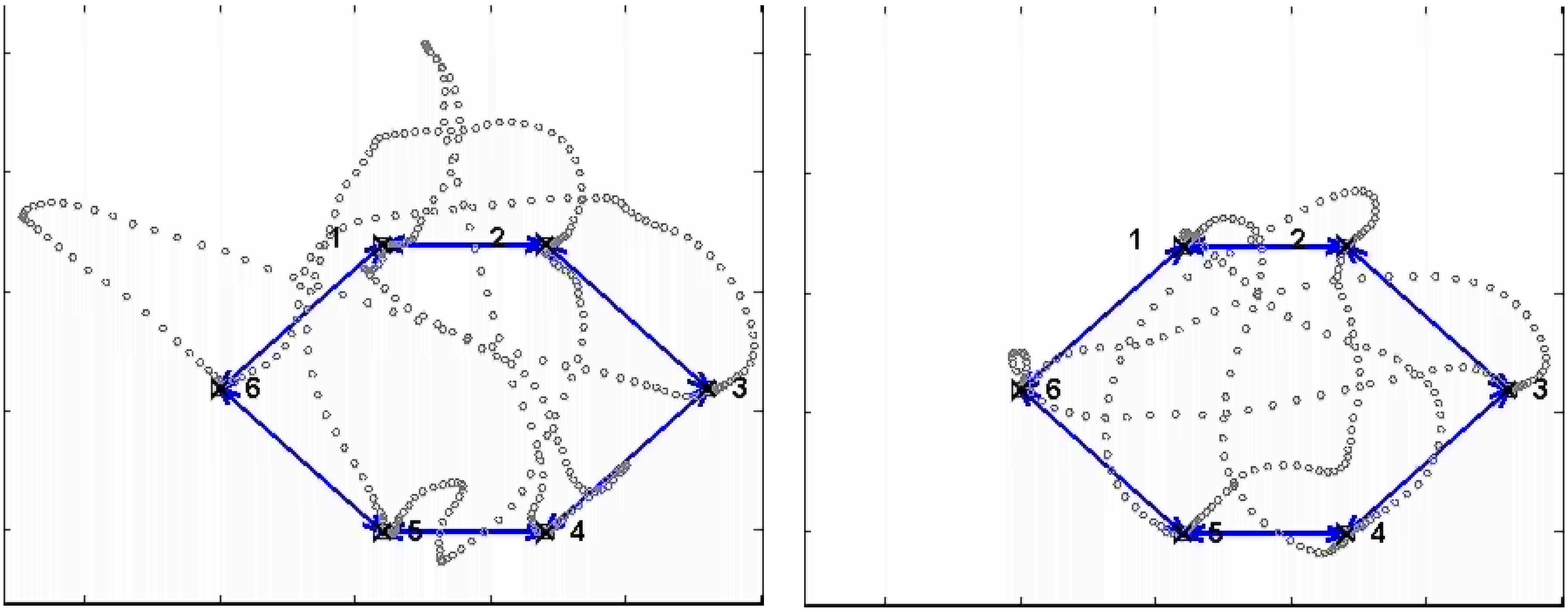
- **Used in cost function**
  Trajectories which do not respect rules can be penalized compared to trajectories enforcing the rules.

- **Used in constraints**
  The local domain of feasibility is reduced to the domain where only trajectories respecting rules are feasible.

  Rules may also represent state-dependent communication schemes or used for signaling model change to neighbors.

- **Crucial assumption**
  Each component has to abide by the same or similar set of rules.

6-Vehicle Formation



Complex Simulation Example

Objective

Autonomous arrangement of a set of OAVs
large and **tight** Formations

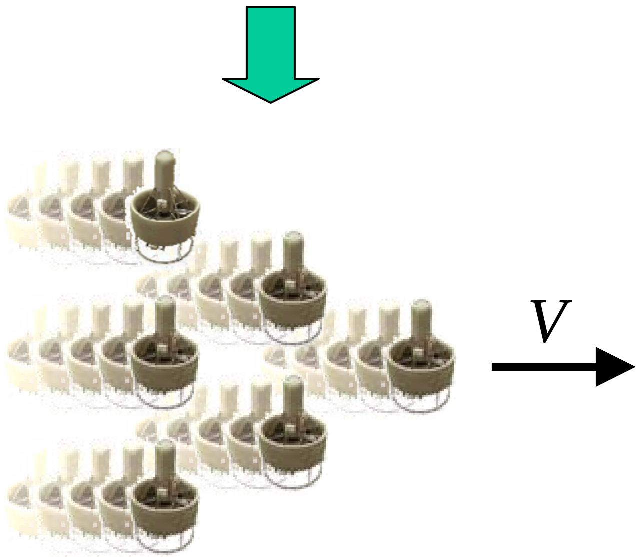
Model

Large scale, piecewise linear models,
dynamically decoupled

Constraints

A: Speed and acceleration constraints on OAV
B: Collision avoidance constraints (non-convex)

6-vehicle formation



Acknowledgements

Prof. Francesco Borrelli
Università del Sannio in Benevento, Italy

Kingsley Fregene
Honeywell Laboratories, Minneapolis, USA

Bálint Vanek
University of Minnesota, Minneapolis, USA