EiR/CPR/CAMS 2024/2025

Analysis and Control of Multi-Robot Systems

Formation Control of Multiple Robots

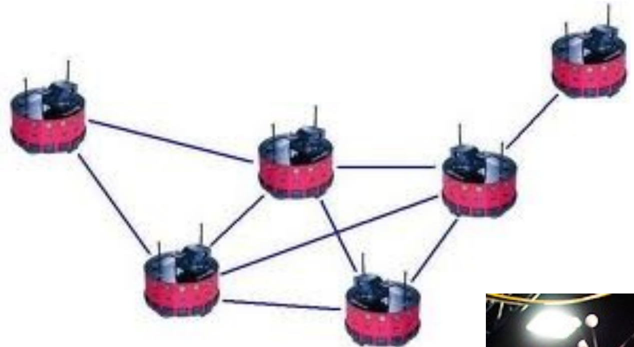
Andrea Cristofaro

(Part of the slides by Paolo Robuffo Giordano)

Dipartimento di Ingegneria Informatica Automatica e Gestionale Antonio Ruberti

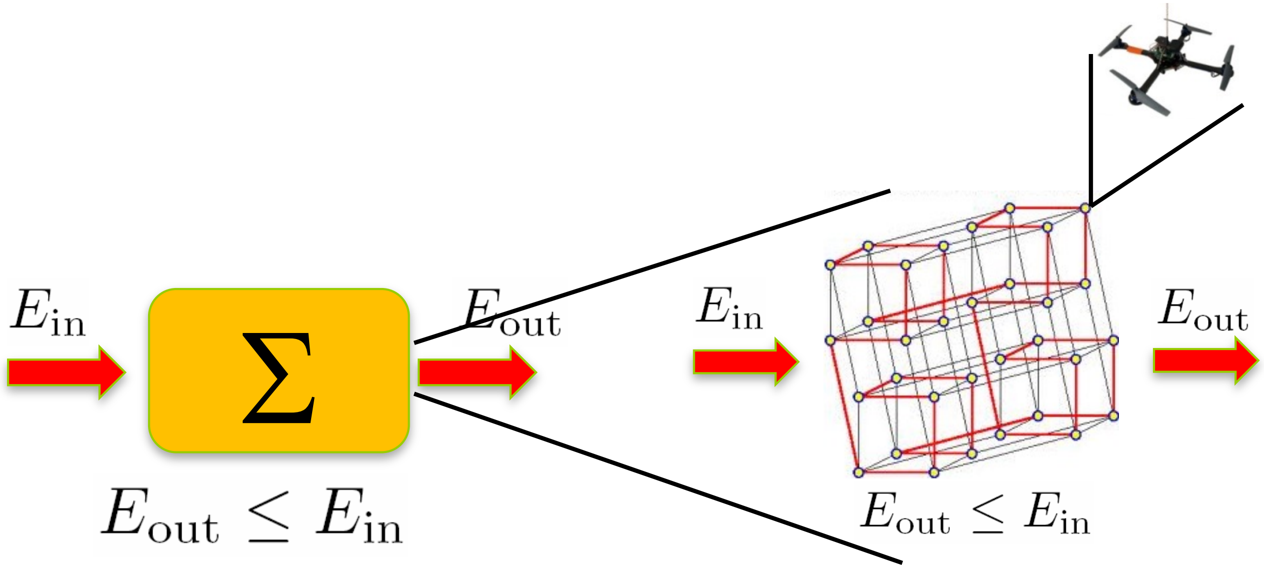
Sapienza Università di Roma

Formation Control of Multiple Robots

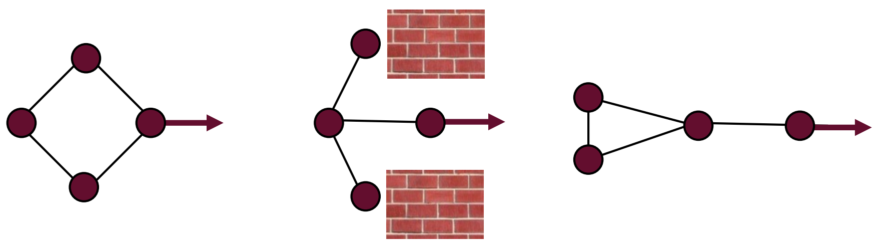


Formation Control of Multiple UAVs

- Let us then focus on the Passivity-based Decentralized Control of Multiple UAVs



- We start with a basic problem: formation control under sensing/comm. constraints

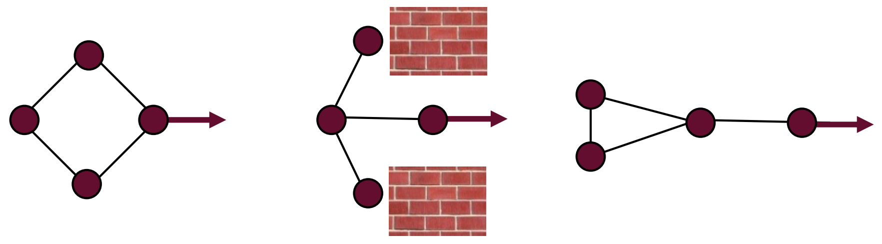


Formation Control of Multiple UAVs

- Formation Control with Time-varying graph topology

- Robots are loosely coupled together
  - can gain/lose neighbors, but must show some form of cohesive behavior

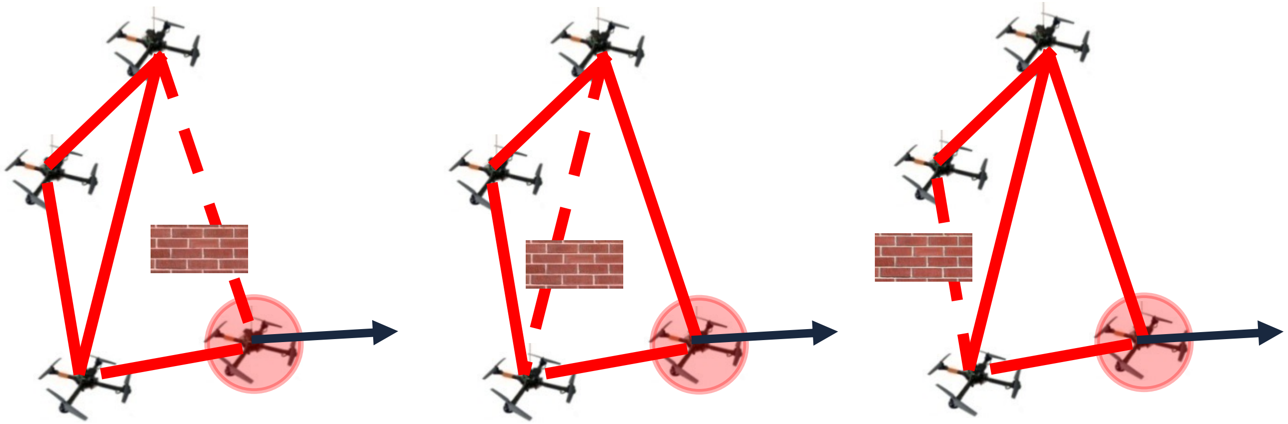
- Robots can decide to split or to join because of any constraint or task, e.g.
  - sensing and/or communication constraints
  - need to temporarily split for better maneuvering in cluttered environments



- Overall motion controlled by selected robots (leaders)

- Appropriate for "loose" tasks, e.g., coverage, persistent patrolling, exploration, etc.

Formation Control of Multiple UAVs



- Features:
  - decentralized design (local and 1-hop communication/sensing)
  - flexible formation: splits/joins due to
    - **sensing/communication constraints**
    - execution of **extra tasks** in parallel to the collective motion
  - Autonomy in **avoiding obstacles** and **inter-agent collisions**

- Challenges:
  - Time-varying topology: ensure stability despite a switching dynamics
  - Guarantee passivity of the overall group behavior
  - Steady-state characteristics? (Velocity synchronization)
  - What if time delays are present in the communication links?
  - What about maintenance of group connectivity?

Agent Model

- Every agent is modeled as a free-floating mass in $\mathbb{R}^3$ with Energy $\mathcal{K}_i = \frac{1}{2} p_i^T M_i^{-1} p_i$

$$\boxed{\begin{cases} \dot{p}_i = F_i^a + F_i^e - B_i M_i^{-1} p_i \\ v_i = \frac{\partial \mathcal{K}_i}{\partial p_i} = M_i^{-1} p_i \end{cases}} \qquad i = 1, \ldots, N$$

- $p_i \in \mathbb{R}^3$ is the agent momentum and $v_i \in \mathbb{R}^3$ the agent velocity. Let also $x_i \in \mathbb{R}^3$, with $\dot{x}_i = v_i$, be the agent position

- $M_i \in \mathbb{R}^{3\times 3}$ is the agent Inertia matrix

- $B_i \geq 0 \in \mathbb{R}^{3\times 3}$ is a velocity damping term (either naturally present or artificially added)

- Force (input) $F_i^a \in \mathbb{R}^3$ represents the interaction (coupling) with the other agents

- Force (input) $F_i^e \in \mathbb{R}^3$ represents the interaction with the “external world” (e.g., obstacles)

Agent Model

- **Remarks:**

- In PHS terms, an agent represents an atomic element storing kinetic energy

$$\mathcal{K}_i = \frac{1}{2} p_i^T M_i^{-1} p_i$$

and endowed with two power ports $(F_i^a,\, v_i)$ and $(F_i^e,\, v_i)$

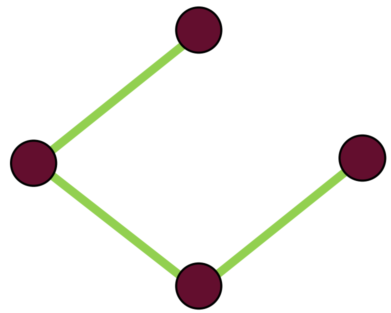
- We consider a simple "free-floating mass" mainly for easiness of exposition
  - Any other (more complex) mechanical (PHS) system would do the job, also constrained (e.g., ground robots)

- The Inertia matrix M_i can model different inertial properties in space
  - e.g., a quadrotor UAV with a faster vertical dynamics w.r.t. the horizontal one

- Heterogeneity in the group can be enforced by choosing different M_i and B_i

Neighboring Definition

- We want to allow for autonomous (and arbitrary) **split/join** decisions because of
  - Sensing/communication constraints
  - Any additional internal criterion (task)



- Let $d_{ij} = \|x_i - x_j\|$ be the interdistance among two agents

- We assume (as usual) presence of a maximum sensing/communication range $D \in \mathbb{R}^+$

- Two agents cannot be neighbors if $d_{ij} > D$ (they cannot sense/communicate with each other)

- To also take into account more general requirements, we introduce a time-varying neighboring condition $\sigma_{ij} \in \{0, 1\}$ satisfying at least:

$$
\begin{aligned}
&1) \quad \sigma_{ij}(t) = 0, \ \textit{if} \ d_{ij} > D \in \mathbb{R}^+; \\
&2) \quad \sigma_{ij}(t) = \sigma_{ji}(t).
\end{aligned}
$$

Neighboring Definition

$$
\begin{aligned}
&1)\quad \sigma_{ij}(t) = 0,\ \textit{if}\ d_{ij} > D \in \mathbb{R}^+;\\
&2)\quad \sigma_{ij}(t) = \sigma_{ji}(t).
\end{aligned}
$$

- Interpretation:
  - Two agents cannot be neighbors if they are too far apart ($d_{ij} > D$)
  - The neighboring condition is symmetric $\sigma_{ij}(t) = \sigma_{ji}(t)$
  - Still, complete freedom in gaining/losing neighbors when $d_{ij} \leq D$
    - Additional sensing/comm. constraints
    - Additional parallel tasks
- This neighboring relationship induces a time-varying Undirected Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}(t))$ where

$$\mathcal{E}(t) = \{(i,j) \in \mathcal{V} \times \mathcal{V} \,|\, \sigma_{ij}(t) = 1 \Leftrightarrow j \in \mathcal{N}_i\}$$

Agent Interconnection

- When neighbors, the agents should keep a cohesive formation

- We consider the (simple) case of maintaining a desired interdistance $0 < d_0 < D$
  - Other more complex (e.g., relative position) cases are possible

$$\begin{cases} \dot{p}_i = F_i^a + F_i^e - B_i M_i^{-1} p_i \\ v_i = \frac{\partial \mathcal{K}_i}{\partial p_i} = M_i^{-1} p_i \end{cases}$$

- This cohesive motion must be achieved by means of local and 1-hop information (decentralization), and by exploiting the coupling force F_i^a in the agent dynamics

- When non-neighbors, no interaction among the agents

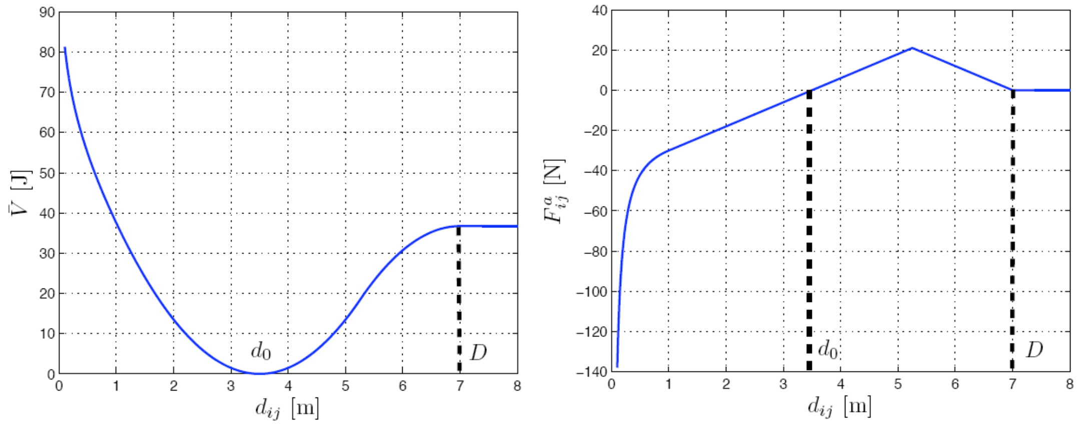
Agent Interconnection

- How to model this interagent coupling? Let us model it as a (nonlinear) elastic element

- Let $x_{ij} \in \mathbb{R}^3$ be the state of this element, and $V(x_{ij}) = \bar{V}(\|x_{ij}\|) \geq 0$ some (lower-bounded) Energy function (Hamiltonian)

- Take the usual PHS form for a storing element $\begin{cases} \dot{x}_{ij} = v_{ij} \\ F^a_{ij} = \dfrac{\partial V(x_{ij})}{\partial x_{ij}} \end{cases}$ where $v_{ij},\ F^a_{ij} \in \mathbb{R}^3$ are the input/output vectors

- For $V(x_{ij})$, we take a function
  - lower-bounded
  - with a minimum at d_0
  - becoming flat for $d_{ij} > D$
  - growing unbounded for $d_{ij} \to 0$



$\bar{V}$ F^a_{ij}

Agent Interconnection

- Say i and j are neighbors, i.e., $\sigma_{ij}(t) = 1 \Leftrightarrow j \in \mathcal{N}_i$, how are they coupled with the elastic element?

$$\begin{cases} \dot{p}_i = F_i^a + F_i^e - B_i M_i^{-1} p_i \\ v_i = \frac{\partial K_i}{\partial p_i} = M_i^{-1} p_i \end{cases} \qquad \begin{cases} \dot{x}_{ij} = v_{ij} \\ F_{ij}^a = \frac{\partial V(x_{ij})}{\partial x_{ij}} \end{cases} \qquad \begin{cases} \dot{p}_j = F_j^a + F_j^e - B_j M_j^{-1} p_j \\ v_j = \frac{\partial K_j}{\partial p_j} = M_j^{-1} p_j \end{cases}$$

- Power preserving interconnection (assume for simplicity everything in $\mathbb{R}$)

$$\begin{bmatrix} F_i^a \\ F_j^a \\ v_{ij} \end{bmatrix} = \boxed{\begin{bmatrix} 0 & 0 & -\sigma_{ij}(t) \\ 0 & 0 & \sigma_{ij}(t) \\ \sigma_{ij}(t) & -\sigma_{ij}(t) & 0 \end{bmatrix}} \begin{bmatrix} v_i \\ v_j \\ F_{ij}^a \end{bmatrix}$$

- Motivation: **when neighbors** $(\sigma_{ij} = 1)$ — **when non-neighbors** $(\sigma_{ij} = 0)$

When neighbors $(\sigma_{ij} = 1)$:

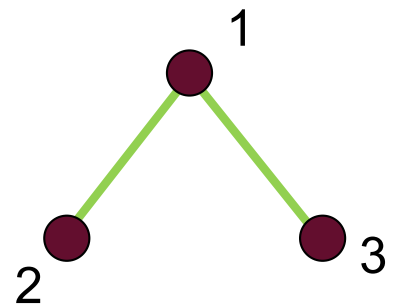
$$v_{ij} = \dot{x}_i - \dot{x}_j = v_i - v_j$$
$$F_i^a = -F_{ij}^a$$
$$F_j^a = F_{ij}^a = -F_{ji}^a$$

When non-neighbors $(\sigma_{ij} = 0)$:

$$v_{ij} = 0$$
$$F_i^a = 0$$
$$F_j^a = 0$$

Agent Interconnection

- Note: for N agents, there exist $N(N-1)/2$ elastic elements (all the possible edges)



- Let us analyze the case of 3 agents with this interaction graph (3 agents and a total of 3 elastic elements)

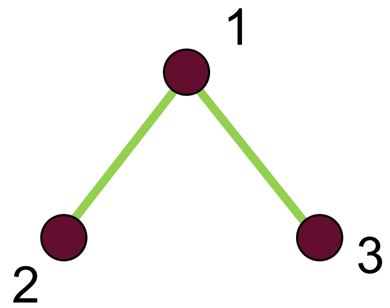
$$\begin{bmatrix} F_1^a \\ F_2^a \\ F_3^a \\ v_{12} \\ v_{13} \\ v_{23} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ F_{12}^a \\ F_{13}^a \\ F_{23}^a \end{bmatrix}$$

Missing edge “23”

Agent Interconnection

- What is the highlighted matrix?

$$\begin{bmatrix} F_1^a \\ F_2^a \\ F_3^a \\ v_{12} \\ v_{13} \\ v_{23} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ F_{12}^a \\ F_{13}^a \\ F_{23}^a \end{bmatrix}$$



- Let us call it $E_{\mathcal{G}} \in \mathbb{R}^{N \times N(N-1)/2}$. This is (almost) the Incidence matrix of graph $\mathcal{G}$
  - labeling and orientation induced by the entries $(v_{12},\ v_{13},\ v_{23})$
  - however, also accounts (with zero columns) for all the missing edges

$$\begin{bmatrix} F_1^a \\ F_2^a \\ F_3^a \\ v_{12} \\ v_{13} \\ v_{23} \end{bmatrix} = \begin{bmatrix} 0 & E_{\mathcal{G}} \\ -E_{\mathcal{G}}^T & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ F_{12}^a \\ F_{13}^a \\ F_{23}^a \end{bmatrix}$$

Agent Interconnection

- Note that $F_i^a = \sum_{j \in \mathcal{N}_i} e_{ij} F_{ij}^a := \boxed{\sum_{j \in \mathcal{N}_i} e_{ij} \frac{\partial \bar{V}}{\partial d_{ij}} \frac{\partial d_{ij}}{\partial x_{ij}}}$

- Therefore, the coupling Force F_i^a for agent i can be computed in a decentralized way
  - Need to know only x_i and $x_j,\ j \in \mathcal{N}_i$

- Let us now generalize for N agents

- Let $x = (x_{12}^T, \dots, x_{1N}^T, x_{23}^T, \dots, x_{2N}^T, \dots, x_{N-1N}^T)^T \in \mathbb{R}^{\frac{3N(N-1)}{2}}$ collect all the elastic element states (edges), and implicitly defining an orientation for the graph $\mathcal{G}$ (labeling and orientation given by the entries in x)

- Let $p = (p_1^T, \dots, p_N^T)^T \in \mathbb{R}^{3N}$ collect all the agent states (momenta)

- Let $B = diag(B_i) \in \mathbb{R}^{3N \times 3N}$ collect all the damping terms

- Let $H = \sum_{i=1}^{N} \mathcal{K}_i + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V(x_{ij})$ be the Total Energy (Hamiltonian)

Agent Interconnection

- The overall group of interconnected agents becomes the **PHS**

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} = \left[\begin{pmatrix} 0 & E(t) \\ -E^T(t) & 0 \end{pmatrix} - \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \right] \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} + GF^e \\ v = G^T \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} \end{cases}$$

- Here, $G = \left((I_N \otimes I_3)^T \quad 0^T \right)^T$ and $E(t) = E_{\mathcal{G}}(t) \otimes I_3$

- The symbol $\otimes$ denotes the Kronecker product among matrixes

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1N}B \\ \vdots & \ddots & \vdots \\ a_{N1}B & \dots & a_{NN}B \end{bmatrix}$$

- And with $F^e = (F_1^{eT} \dots F_N^{eT})^T \in \mathbb{R}^{3N}$ being the input and $v = (v_1^T \dots v_N^T) \in \mathbb{R}^{3N}$ the output vectors

Agent Interconnection

- The PHS group of agents

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} = \left[\begin{pmatrix} 0 & E(t) \\ -E^T(t) & 0 \end{pmatrix} - \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \right] \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} + GF^e \\ v = G^T \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} \end{cases}$$

has then an external port $(v,\ F^e)$ where $F^e = (F_1^{eT} \dots F_N^{eT})^T \in \mathbb{R}^{3N}$ and

$v = (v_1^T \dots v_N^T) \in \mathbb{R}^{3N}$

$$\begin{cases} \dot{p}_i = F_i^a + F_i^e - B_i M_i^{-1} p_i \\ v_i = \frac{\partial K_i}{\partial p_i} = M_i^{-1} p_i \end{cases}$$

- The port $(v,\ F^e)$ is the one interacting with the external world (**obstacles**, **external commands**)

- Let us then study the passivity of the group w.r.t. the port $(v,\ F^e)$

Passivity of the Group

- Suppose for now a fixed topology for the graph, i.e., $\boxed{E(t) = E = const}$
- Since H is lower bounded, the group of agents is passive w.r.t. its external port

$$\dot{H} = -\frac{\partial^T H}{\partial p} B \frac{\partial H}{\partial p} + v^T F^e \leq v^T F^e$$

- Does this automatically extend to the general case $E(t)$?
- Consider first the case of a split $\sigma_{ij} = 1 \rightarrow \sigma_{ij} = 0$
- The edge $(i,\ j)$ is lost and the Incidence matrix is updated accordingly $E \rightarrow E'$
- The group dynamics becomes

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \end{pmatrix} = \left[\begin{pmatrix} 0 & E' \\ -E'^T & 0 \end{pmatrix} - \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \right] \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} + G F^e \\ v = G^T \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial x} \end{pmatrix} \end{cases}$$

Passivity of the Group

- Since E' appears in a skew-symmetric matrix, overall passivity is preserved

$$\dot{H} = -\frac{\partial^T H}{\partial p} B \frac{\partial H}{\partial p} + v^T F^e \leq v^T F^e$$

- Then, **exactly the same argument** should work for the join case $\sigma_{ij} = 0 \rightarrow \sigma_{ij} = 1$

- **Unfortunately, it doesn't!!!**

- During a join, the Incidence matrix is updated as before $E \rightarrow E'$

- BUT this is not the only action needed to join

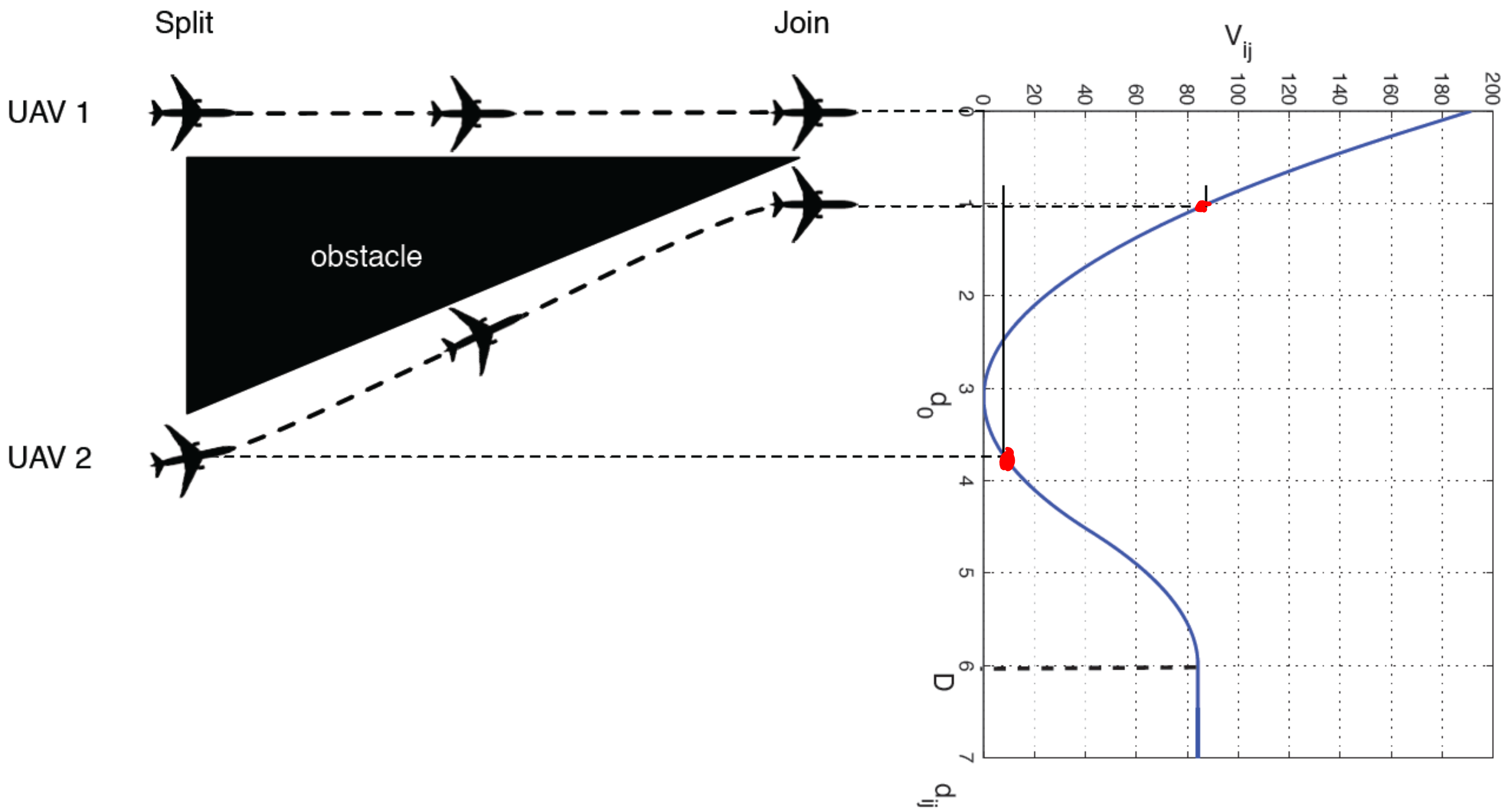
- At the join, the state of the elastic element must be reset to the actual relative position of agents i and j

$$\boxed{x_{ij} \leftarrow x_i - x_j}$$

- This action, in general, **costs extra energy**! (thus, can violate passivity)

Passivity of the Group

- Consider this situation (visibility and interdistance determine neighboring)



- Because of different interdistances at the split and join decisions, it is $V_{\text{join}} > V_{\text{split}}$

Passivity of the Group

- How to still implement a join procedure? How to passify it?

- If some $\Delta V = V_{\text{join}} - V_{\text{split}} > 0$ is needed, this must be drawn from energy sources already present in the agent group
  - Passivity = no internal production of extra energy

- Can we find some internal energy storages from which to cover for ΔV ?

- Make use of Energy Tanks and Energy Transfer control
  - Store back the agent inherent dissipation $D_i = p_i^T M_i^{-T} B_i M_i^{-1} p_i$
  - Exploit Tank Energies for passively implement a (otherwise non-passive) join
  - Obviously, everything still to be done in a decentralized way…

Passivity of the Group

- Let us then apply the Tank machinery

$$T(x_{t_i}) = \frac{1}{2}x_{t_i}^2 \geq 0$$

- First, augment the agent state with the Tank dynamics, with

$$\begin{cases} \dot{p}_i = F_i^a + F_i^e - B_i M_i^{-1} p_i \\ v_i = \frac{\partial K_i}{\partial p_i} = M_i^{-1} p_i \end{cases} \Longrightarrow \begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \frac{1}{x_{t_i}} D_i + w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

- Second, endow the elastic elements with an additional input $w_{ij}^x \in \mathbb{R}^3$ for exchanging energy with the Tanks

$$\begin{cases} \dot{x}_{ij} &= v_{ij} \\ F_{ij}^a &= \frac{\partial V(x_{ij})}{x_{ij}} \end{cases} \Longrightarrow \begin{cases} \dot{x}_{ij} &= v_{ij} + w_{ij}^x \\ F_{ij}^a &= \frac{\partial V(x_{ij})}{x_{ij}} \end{cases}$$

Passivity of the Group

$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \dfrac{1}{x_{t_i}} D_i + w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases} \Longleftrightarrow \begin{cases} \dot{x}_{ij} &= v_{ij} + w_{ij}^x \\ F_{ij}^a &= \dfrac{\partial V(x_{ij})}{x_{ij}} \end{cases} \Longleftrightarrow \begin{cases} \dot{p}_j &= F_j^a + F_j^e - B_j M_j^{-1} p_j \\ \dot{x}_{t_j} &= \dfrac{1}{x_{t_j}} D_j + w_{ji}^t \\ y &= \begin{bmatrix} v_j \\ x_{t_j} \end{bmatrix} \end{cases}$$

- Exploiting the Energy Transfer control:
  - inputs w_{ij}^x , w_{ij}^t and w_{ji}^t will allow for drawing ΔV from the Tanks of agents i and j
  - this allows implementing the join action (and resetting the spring state to the correct value $x_{ij} \leftarrow x_i - x_j$)

- Recall, the Energy Transfer control among two **PHS** was implemented by the coupling

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 & -\alpha y_1(x_1) y_2^T(x_2) \\ \alpha y_2(x_2) y_1^T(x_1) & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad \alpha \in \mathbb{R} \qquad x_{t_j}$$

- Likewise, we choose $\begin{bmatrix} w_{ij}^x \\ w_{ij}^t \\ w_{ji}^t \end{bmatrix} = \begin{bmatrix} 0 & -\gamma_{ij} F_{ij}^a x_{t_i} & -\gamma_{ij} F_{ij}^a x_{t_j} \\ \gamma_{ij} {F_{ij}^a}^T x_{t_i} & 0 & 0 \\ \gamma_{ij} {F_{ij}^a}^T x_{t_j} & 0 & 0 \end{bmatrix} \begin{bmatrix} F_{ij}^a \\ x_{t_i} \\ x_{t_j} \end{bmatrix}, \ \gamma_{ij} \in \mathbb{R}$

Passivity of the Group

- The parameter γ_{ij} dictates the rate and direction of Energy transfer
- A value $\gamma_{ij} < 0$ refills the spring energy and draws from the two Tanks
- The machinery easily extends to multiple connections among agents and springs

$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \dfrac{1}{x_{t_i}} D_i + \boxed{\sum_{j=1}^N w_{ij}^t} \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

- Note that the previous interconnection can be implemented in a decentralized way
  - agent i needs to know F_{ij}^a and t_i (local and 1-hop information)
  - $\gamma_{ij} = 0$ by convention if $j \notin \mathcal{N}_i$

Passivity of the Group

- Strategy for implementing a join decision in a passive way among agents $(i,\ j)$:
  - 1. at the join moment, compute $\Delta V = V(x_i - x_j) - V(x_{ij})$
  - 2. if $\Delta V \leq 0$, implement the join (and store ΔV back into the tanks T_i and T_j)
  - 3. if $\Delta V > 0$, extract ΔV from T_i and T_j

- **What if $T_i + T_j < \Delta V$?**

- Must take a decision:
  - Do not join (and wait for better conditions)
  - Ask the rest of the group for "help"

- How to ask for "help" in a decentralized and passive way?
  - **A possibility**: run a **consensus** on all the Tank Energies
  - This redistributes the energies within the group
  - But it doesn't change the total amount of energy

Passivity of the Group

- Additional (and last) modification to the agent dynamics

$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \boxed{(1-\beta_i)} \left(\frac{1}{x_{t_i}} D_i + \sum_{j=1}^{N} w_{ij}^t \right) \boxed{+\, \beta_i c_i} \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

- The parameter $\beta_i \in \{0,\ 1\}$ enables/disables the consensus mode

- During consensus ($\beta_i = 1$), we want $\dot{T}_i = -\sum_{j \in \mathcal{N}_i} (T_i - T_j)$

- This is achieved by setting $\boxed{c_i = -\frac{1}{t_i} \sum_{j \in \mathcal{N}_i} (T_i(t_i) - T_j(t_j))}$

Passivity of the Group

- Compact form of the Passive Join procedure (decentralized and passive)

Procedure PassiveJoin

Data: $x_i, x_j, x_{ij}^s, t_i, t_j$

1 *Compute* $\Delta E = V(x_i - x_j) - V(x_{ij}^s)$;

2 **if** $\Delta E \leq 0$ **then**

3   *Store* $(-\Delta E)/2$ in the tank through input w_{ij};

else

4   **if** $T_i(t_i) + T_j(t_j) < \Delta E + 2\varepsilon$ **then**

5   Run a *consensus* on the tank variables;

6   **if** $2T_i(t_i) < \Delta E + 2\varepsilon$ **then**

7   *Dampen* until $T(t_i) + T(t_j) \geq \Delta E + 2\varepsilon$;

8   *Extract* $\dfrac{T(t_i)}{T(t_i) + T(t_j)} \Delta E$ from the tank through input w_{ij};

9 *Join*;

- Note: if after the consensus still not enough energy (line 6)
  - The agents do not join
  - They can switch to a high damping mode for more quickly refilling the Tanks

Passivity of the Group

- By considering the Tank dynamics and the PassiveJoin Procedure, the group dynamics becomes

$$\begin{cases} \begin{bmatrix} \dot{p} \\ \dot{x} \\ \dot{x}_t \end{bmatrix} = \left(\begin{bmatrix} 0 & E(t) & 0 \\ -E^T(t) & 0 & \Gamma^T \\ 0 & -\Gamma & 0 \end{bmatrix} - \begin{bmatrix} B & 0 & 0 \\ 0 & 0 & 0 \\ -(I-\beta)PB & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial x_t} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \beta c \end{bmatrix} + GF^e \\ v = G^T \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial x_t} \end{bmatrix} \end{cases}$$

where the new Hamiltonian is $\mathcal{H} = \sum_{i=1}^{N} \mathcal{K}_i + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V(x_{ij}) \boxed{+ \sum_{i=1}^{N} T_i}$ and $\beta = diag(\beta_i)$, $P = diag(\frac{1}{t_i} p_i^T M_i^{-T})$, and matrix $\Gamma \in \mathbb{R}^{N \times \frac{3N(N-1)}{2}}$ representing the interconnection between Tanks and springs

Passivity of the Group

- Proposition: the group dynamics (with Tanks, Energy Transfer, Consensus, and PassiveJoin Procedure)

$$\begin{cases} \begin{bmatrix} \dot{p} \\ \dot{x} \\ \dot{x}_t \end{bmatrix} = \left(\begin{bmatrix} 0 & E(t) & 0 \\ -E^T(t) & 0 & \Gamma^T \\ 0 & -\Gamma & 0 \end{bmatrix} - \begin{bmatrix} B & 0 & 0 \\ 0 & 0 & 0 \\ -(I-\beta)PB & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial x_t} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \beta c \end{bmatrix} + GF^e \\ v = G^T \begin{bmatrix} \frac{\partial \mathcal{H}}{\partial p} \\ \frac{\partial \mathcal{H}}{\partial x} \\ \frac{\partial \mathcal{H}}{\partial x_t} \end{bmatrix} \end{cases}$$

is still **passive** $\boxed{\dot{\mathcal{H}} \leq v^T F^e}$

- **Proof: left as exercise**

Passivity of the Group

- Additional remarks:

- We can always enforce a limiting strategy for the Tank refilling action by means of a parameter $\alpha_i \in \{0,\ 1\}$

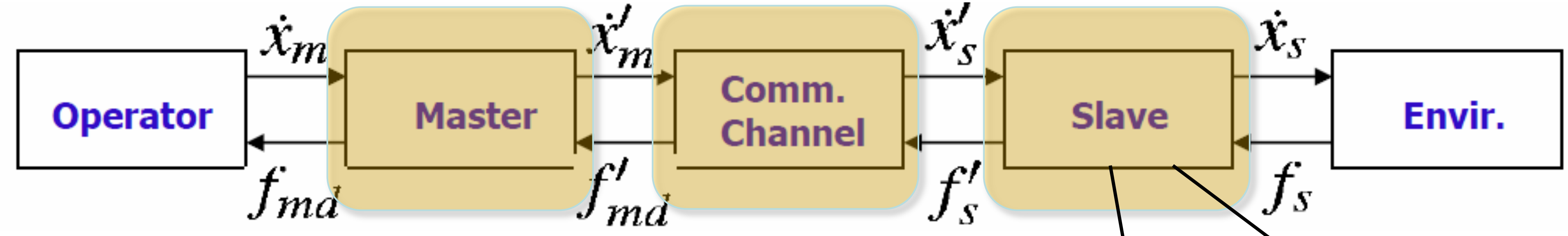
$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \boxed{\alpha_i} \dfrac{1}{x_{t_i}} D_i + \sum_{j=1}^{N} w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

such that $\alpha_i = \begin{cases} 0, & \text{if} \quad T_i \geq \bar{T}_i \\ 1, & \text{if} \quad T_i < \bar{T}_i \end{cases}$ where $\bar{T}_i$ is a suitable upper bound for the Tank energy level

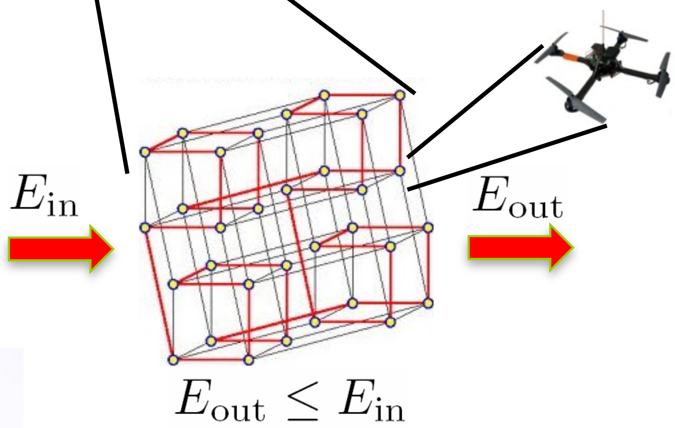
- This way, we can avoid a too large and unnecessary accumulation

Bilateral Teleoperation of Multiple UAVs

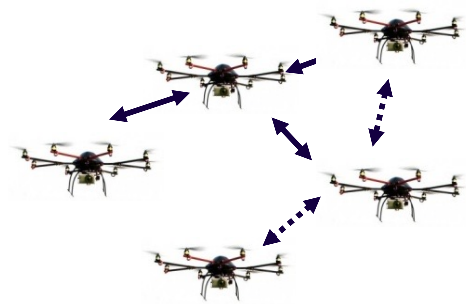
- Let us see an application of this framework: the case of Bilateral Teleoperation



- Thanks to its passivity, we can easily attach the agent group (slave-side) to the master-side of a bilateral teleoperation system



- **Control** subset of UAV DOFs
  - "High-level" motion commands
- **Receive** a useful force feedback
  - number of agents in the group
  - speed of agents
  - interaction with the environment



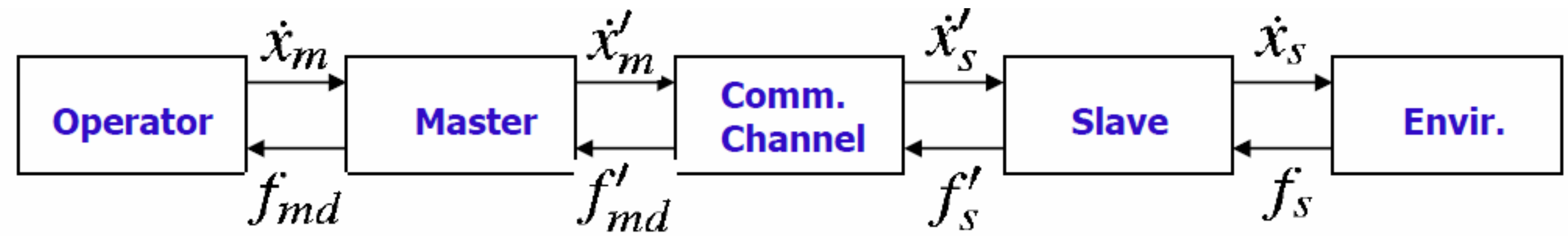
Bilateral Teleoperation

- Components of a Bilateral Teleoperation System
  - Operator (Human) acts on the Master (and vice-versa)
  - Master transmits and receives info via a comm. channel
  - Slave transmits and receives info via a comm. channel
  - Environment acts on the Slave (and vice-versa)

- Everything can be modeled as an exchange of force/position information…

- ….that is, exchange of power variables (energy)

- Indeed, every component can be modeled as a N-port PHS

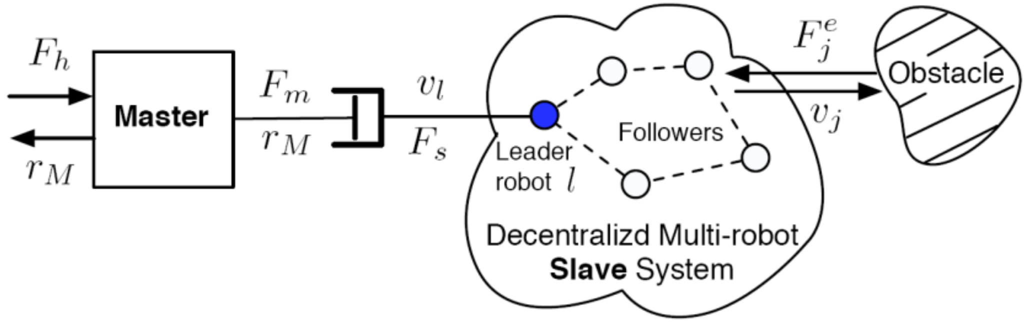


Bilateral Teleoperation of Multiple UAVs

- Consider one leader, and split its external force as $F_l^e = F_s + F_l^{\text{env}}$

- Interconnect master and leader in this (passive) way

$$\begin{cases} F_s = b_T(r_M - v_l) \\ F_m = -b_T(r_M - v_l) \end{cases}, \quad b_T > 0$$



- v_l is the leader velocity
- r_M is (almost) the master configuration, corresponding to a velocity command

- Force F_m will inform about the mismatch $v_l - r_M$
  - Number of agents in the connected component of the leader (their total inertia)
  - Absolute speed of the group
  - Interaction with the environment (obstacles)

- Obstacles are considered as passive systems producing repulsive forces (spring-like elements)

Velocity Synchronization

- Assume a constant velocity command for the leader $r_M = const$

- Do the agents, at steady-state, synchronize with this velocity command?

$$\boxed{v_i \to r_M, \quad \forall i\ ?}$$

- We must characterize the steady-state of the system (**<u>if it exists</u>**)

- Assumptions for the steady-state:
  - 1) $F_i^{\text{env}} = 0, \quad \forall i = 1, \dots, N$ (no environmental forces - no close obstacles)
  - 2) Tanks are full to $\bar{T}_i$ and $\Gamma = 0$ (no joins, no energy exchanges with elastic elements)
  - 3) $\mathcal{G}$ is **connected** (can always reduce to the connected component of the leader)

- Also assume (w.l.o.g.) that the leader is agent 1
  - For the leader, $F_1^e = F_s = b_T(r_M - v_1)$
  - For all the others, $F_i^e = F_i^{\text{env}} = 0$ (because of Assumption 1)

Velocity Synchronization

- Step 1: with $r_M = const$ and Assumptions 1), 2), 3), prove existence of a steady-state

- It can be proven that a steady-state exists such that $\boxed{(\dot{p},\ \dot{x},\ \dot{t}) = (0,\ 0,\ 0)}$
  - follows from "exosystem"-like arguments + output strictly passivity of the slave-side (see, e.g., Isidori's book Nonlinear Control Systems)

- At steady-state:
  - Velocities stay constant $(\dot{p} = \dot{v} = 0)$ (assuming constant mass for the agents)
  - Spring lengths (relative positions) stay constant $(\dot{x} = 0)$
  - Tank energies stay constant $(\dot{t} = 0)$. This follows from Assumption 2)

- To which steady-state velocity do the agents converge? Is it $v_i = r_M = const$ for all of them?

Velocity Synchronization

- Under these assumptions, and splitting F_s into the two contributions $b_T r_M$ and $-b_T v_1$, one can rewrite the agent group dynamics as

$$\begin{pmatrix} \dot{p} \\ \dot{x} \\ \dot{t} \end{pmatrix} = \begin{pmatrix} -B' & E & 0 \\ -E^T & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \dfrac{\partial \mathcal{H}}{\partial p} \\ \dfrac{\partial \mathcal{H}}{\partial x} \\ \dfrac{\partial \mathcal{H}}{\partial t} \end{pmatrix} + \begin{pmatrix} I_{3N} \\ 0 \\ 0 \end{pmatrix} u$$

where $B' = \mathrm{diag}(B_i')$, $B_1' = B_1 + b_T I_3$, $B_i' = B_i$ and $u = \big(b_T r_M^T \, 0 \ldots 0\big)^T \in \mathbb{R}^{3N}$

- And then impose the steady-state condition $(\dot{p},\, \dot{x},\, \dot{t}) = (0,\, 0,\, 0)$

- The first "row" becomes $B' \dfrac{\partial \mathcal{H}}{\partial p} - E \dfrac{\partial \mathcal{H}}{\partial x} = u$

- The second "row" becomes $E^T \dfrac{\partial \mathcal{H}}{\partial p} = 0$

Velocity Synchronization

- Two conditions: $B'\frac{\partial \mathcal{H}}{\partial p} - E\frac{\partial \mathcal{H}}{\partial x} = u$ and $E^T\frac{\partial \mathcal{H}}{\partial p} = 0$

- We know that, for connected graphs, $\ker E^T = \mathbf{1}_{N_3}$ where $\mathbf{1}_{N_3} = \mathbf{1}_N \otimes I_3$

- Therefore, $\boxed{\frac{\partial \mathcal{H}}{\partial p} = \mathbf{1}_{N_3} v_{ss}}$ for some $v_{ss} \in \mathbb{R}^3$. All the agents have the same velocity

- By plugging this result into the first condition, we get $B'\mathbf{1}_{N_3} v_{ss} - E\frac{\partial \mathcal{H}}{\partial x} = u$

- Pre-multiply both sides by $\mathbf{1}_{N_3}^T$ to get $\mathbf{1}_{N_3}^T B'\mathbf{1}_{N_3} v_{ss} = \mathbf{1}_{N_3}^T u = b_T r_M$

- This finally results into the sought value $\boxed{v_{ss} = (\mathbf{1}_{N_3}^T B'\mathbf{1}_{N_3})^{-1} b_T r_M}$

Velocity Synchronization

- Conclusions: at steady-state (Assumptions 1), 2), 3) and $r_M = const$), all the agent velocities reach

$$\boxed{v_i \to v_{ss} = (\mathbf{1}_{N_3}^T B' \mathbf{1}_{N_3})^{-1} b_T r_M}$$

- One can check that $\|v_{ss}\| < \|r_M\|$

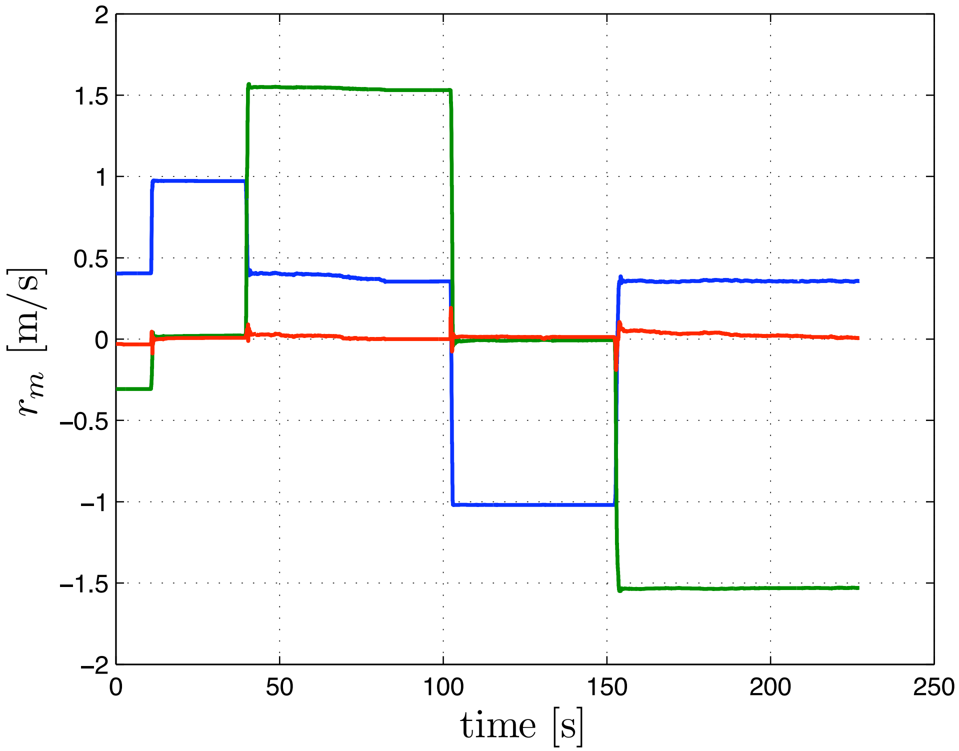
- For instance, for "scalar" damping terms $B_i = b_i I_3$ this reduces to

$$\boxed{v_{ss} = \frac{b_T r_M}{b_T + \sum b_i}}$$

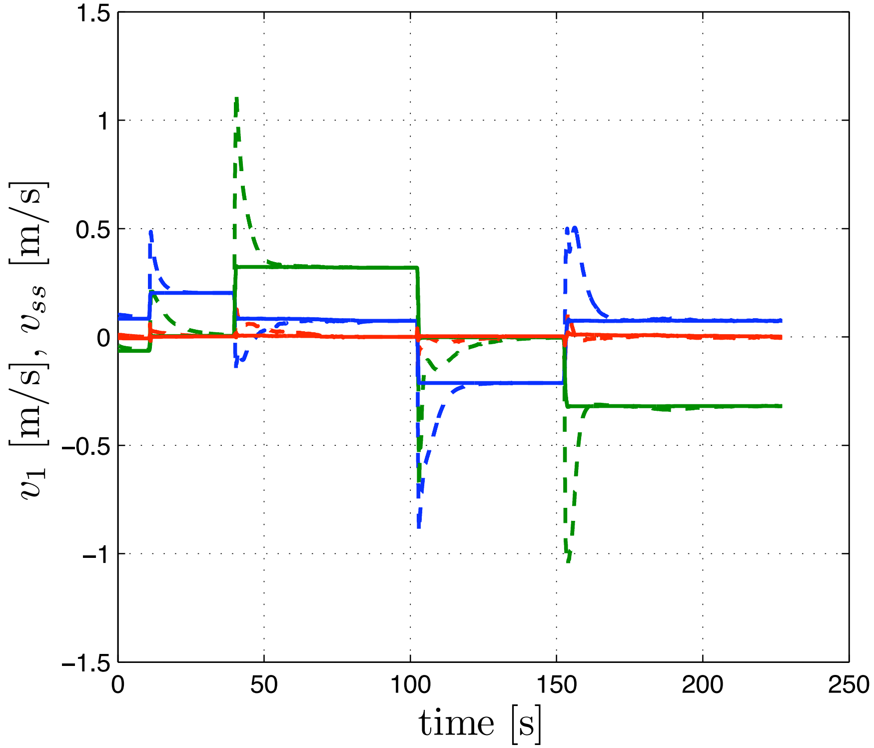
- The agents always travel "slower" than the commanded r_M

- Perfect synchronization only if $b_i = 0$ (no damping on any agent!)

Velocity Synchronization

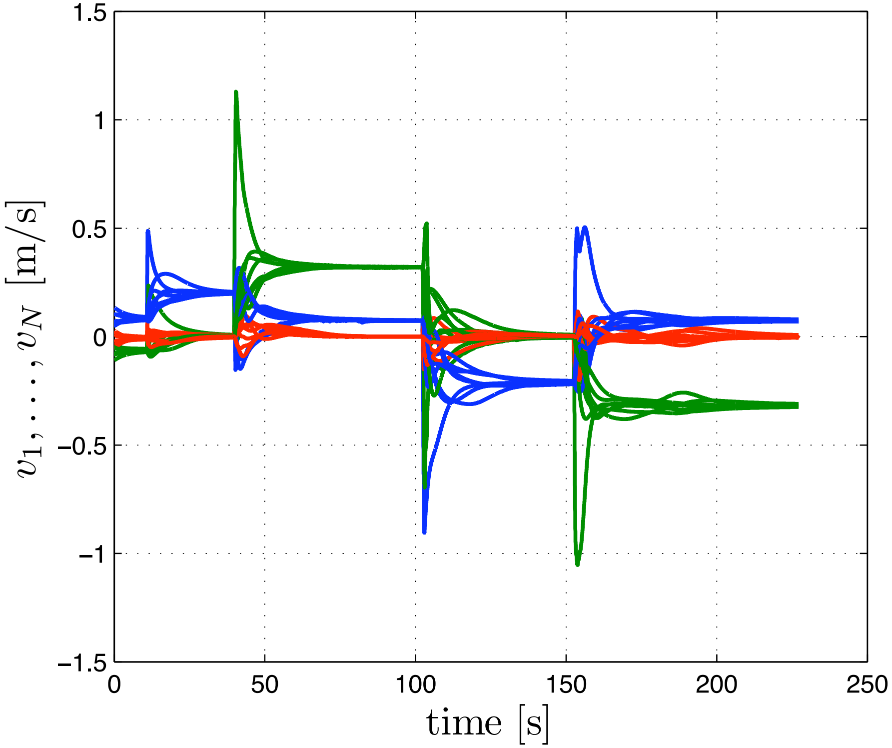


Leader velocity command r_M

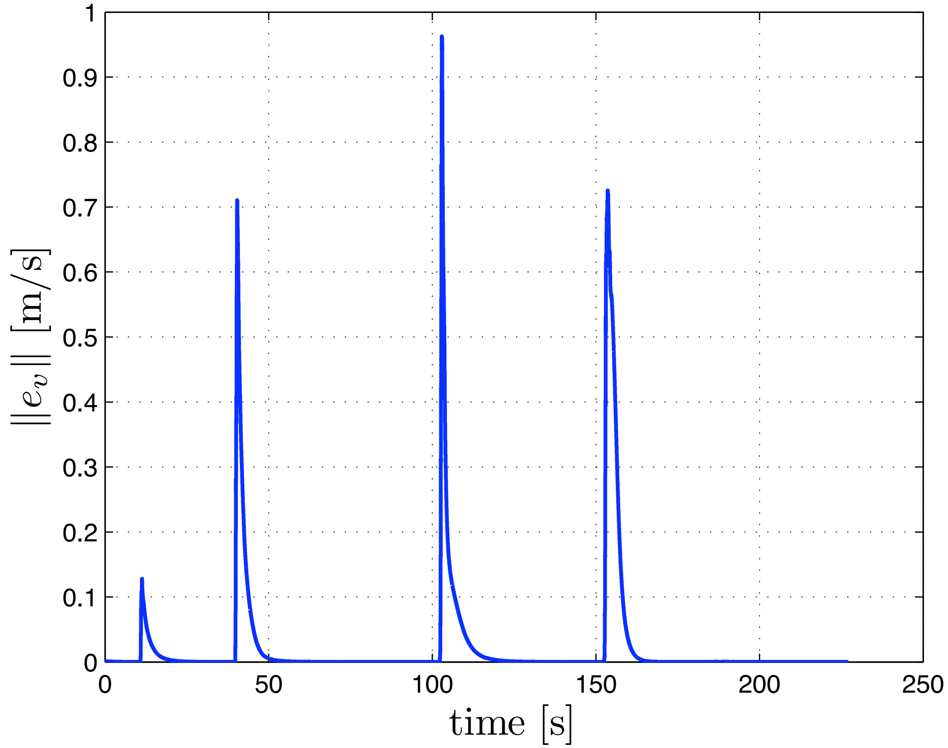


Leader vel. v_1 vs. predicted v_{ss}

Velocity Synchronization



All agent velocities



Norm of velocity synchronization error

$$\|e_v\| = \|v - \mathbf{1}_{N_3} v_{ss}\|$$

Velocity Synchronization

- How to synchronize velocities with r_M (at steady-state)?

- The damping terms B_i are
  - good for stabilization and Tank refill
  - bad for vel. synchronization, as they "slow down" the agents….
  - ….it seems they should be "switched off"

- Must modify the agent dynamics: consider

$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} &= \dfrac{1}{x_{t_i}} D_i + \sum_{j=1}^N w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases} \Longrightarrow \begin{cases} \dot{p}_i &= F_i^a + F_i^e + F_i^s + F_i^d \\ \dot{x}_{t_i} &= \dfrac{1}{x_{t_i}} D_i + \sum_{j=1}^N w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

where $F_i^d = -B_i(t_i) M_i^{-1} p_i$ is the "damping" force, but with a **variable damping term**

$$\boxed{B_i(t_i) = \begin{cases} 0 & \text{if} \quad T(t_i) = \bar{T}_i \\ \bar{B}_i & \text{if} \quad T(t_i) < \bar{T}_i \end{cases}}$$

Velocity Synchronization

- The damping B_i is now active only when needed to refill the Tank T_i

$$\begin{cases} \dot{p}_i &= F_i^a + F_i^e + F_i^s + F_i^d \\ \dot{x}_{t_i} &= \dfrac{1}{x_{t_i}} D_i + \sum_{j=1}^N w_{ij}^t \\ y &= \begin{bmatrix} v_i \\ x_{t_i} \end{bmatrix} \end{cases}$$

- The additional (synchronization) force F_i^s is designed as $F_i^s = -b \sum_{j \in \mathcal{N}_i} (v_i - v_j)$ (**consensus among velocities**)

- The group dynamics takes the form

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \\ \dot{t} \end{pmatrix} = \left[\begin{pmatrix} 0 & E & 0 \\ -E^T & 0 & \Gamma^T \\ 0 & -\Gamma & 0 \end{pmatrix} - \begin{pmatrix} \mathcal{L} + B & 0 & 0 \\ 0 & 0 & 0 \\ -PB & 0 & 0 \end{pmatrix} \right] \nabla \mathcal{H} + G F^e \\ v = G^T \nabla \mathcal{H} \end{cases}$$

Velocity Synchronization

- Matrix $\mathcal{L} = bL \otimes I_3$ where L is the Laplacian of the Graph $\mathcal{G}$

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x} \\ \dot{t} \end{pmatrix} = \left[\begin{pmatrix} 0 & E & 0 \\ -E^T & 0 & \Gamma^T \\ 0 & -\Gamma & 0 \end{pmatrix} - \begin{pmatrix} \mathcal{L} + B & 0 & 0 \\ 0 & 0 & 0 \\ -PB & 0 & 0 \end{pmatrix} \right] \nabla \mathcal{H} + GF^e \\ v = G^T \nabla \mathcal{H} \end{cases}$$

- Exercise: **prove that the system is passive**

$$\dot{\mathcal{H}} = -\frac{\partial^T \mathcal{H}}{\partial p} \mathcal{L} \frac{\partial \mathcal{H}}{\partial p} + v^T F^e \leq v^T F^e$$

- What can be said about the steady-state regime?

Velocity Synchronization

- Assumptions (analogously to before):
  - 1) $F_i^{\mathrm{env}} = 0, \quad \forall i = 1, \ldots, N$ (no external forces – no close obstacles)
  - 2) $B_i(t_i) = 0$ and $\Gamma = 0$ (tanks full and no energy exchange with elastic elements)
  - 3) $\mathcal{G}$ is **connected**

- Then, at steady-state:
  - 1) $v_i \to r_M = const$ (all agents synchronize with the commanded velocity)
  - 2) $\dot{x} = 0$ (all spring lengths/relative positions stay constant)

- Proof: apply the change of coordinates $(p,\, x,\, t) \to (\tilde{p},\, x,\, t)$ where $\boxed{\tilde{p}_i = p_i - M_i r_M}$
- The quantity $\tilde{p}_i$ is the "**momentum (velocity) synchronization error**"

- New "energy" $\boxed{\tilde{\mathcal{K}}_i = \frac{1}{2}\tilde{p}_i^T M_i^{-1}\tilde{p}_i,}$ $\quad \tilde{H} = \sum_{i=1}^{N}\tilde{\mathcal{K}}_i + \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} V(x_{ij}) + \sum_{i=1}^{N} T_i$

Velocity Synchronization

- What is the dynamics of $\tilde{p}$? $\dot{\tilde{p}} = \dot{p} = E\frac{\partial \mathcal{H}}{\partial x} - (\mathcal{L} + B)\frac{\partial \mathcal{H}}{\partial p} + GF^e$

- **Facts:**
  - 1) $\frac{\partial \mathcal{H}}{\partial x} = \frac{\partial \tilde{\mathcal{H}}}{\partial x}$ and $\boxed{\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = v - \mathbf{1}_{N_3} r_M = \frac{\partial \mathcal{H}}{\partial p} - \mathbf{1}_{N_3} r_M}$
  - 2) $\mathcal{L}\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = \mathcal{L}\frac{\partial \mathcal{H}}{\partial p}$ and $E^T\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = E^T\frac{\partial \mathcal{H}}{\partial p}$
  - 3) $B = 0$ and $\Gamma = 0$ (assumption 2))

- Then, the group dynamics can be rewritten in terms of new coordinates and new energy function

$$\boxed{\dot{\tilde{p}} = E\frac{\partial \tilde{\mathcal{H}}}{\partial x} - \mathcal{L}\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} + GF^e} \qquad \boxed{\dot{x} = -E^T\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}}}$$

Velocity Synchronization

- New dynamics

$$\begin{cases} \begin{pmatrix} \dot{\tilde{p}} \\ \dot{x} \\ \dot{t} \end{pmatrix} = \left[\begin{pmatrix} 0 & E & 0 \\ -E^T & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} \mathcal{L} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right] \nabla \tilde{H} + GF^e \\ v = G^T \nabla \tilde{H} \end{cases}$$

- Let us study the asymptotic stability: $\dot{\tilde{H}} = -\frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}} \mathcal{L} \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} + \frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}} F^e$

- By using Assumption 1) and the expression of $F_s = b_T(r_M - v_1) = -b_T \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}_1}$

we obtain $\dot{\tilde{\mathcal{H}}} = -\frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}} \mathcal{L} \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} - \boxed{\frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}_1} b_T \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}_1}} \leq 0$

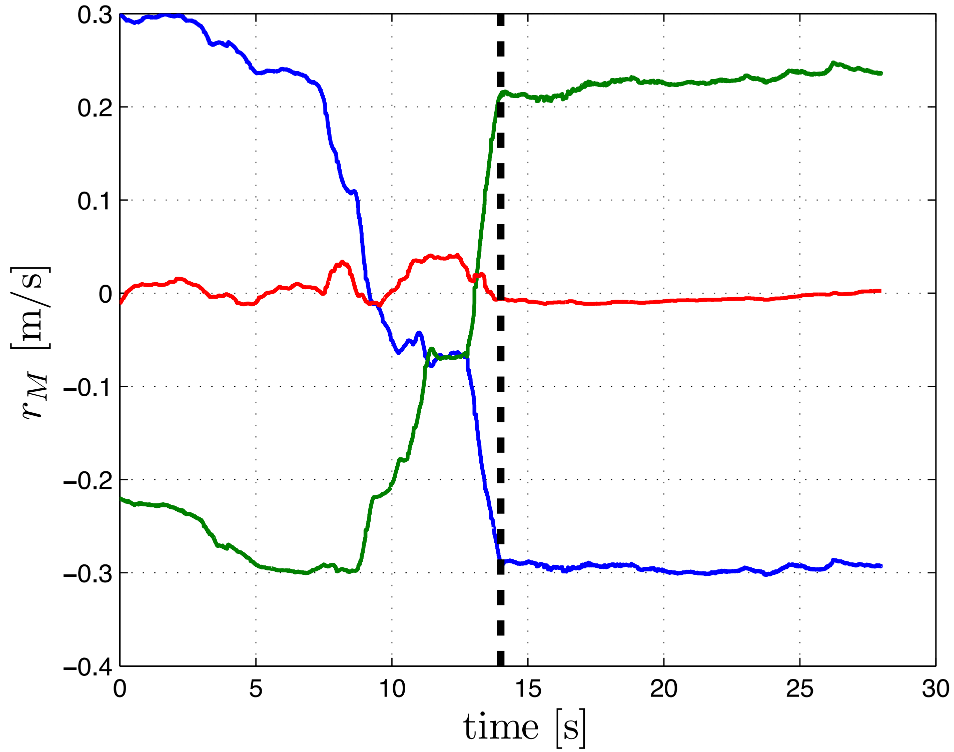
- Energy function non-increasing -> system trajectories are bounded

Velocity Synchronization

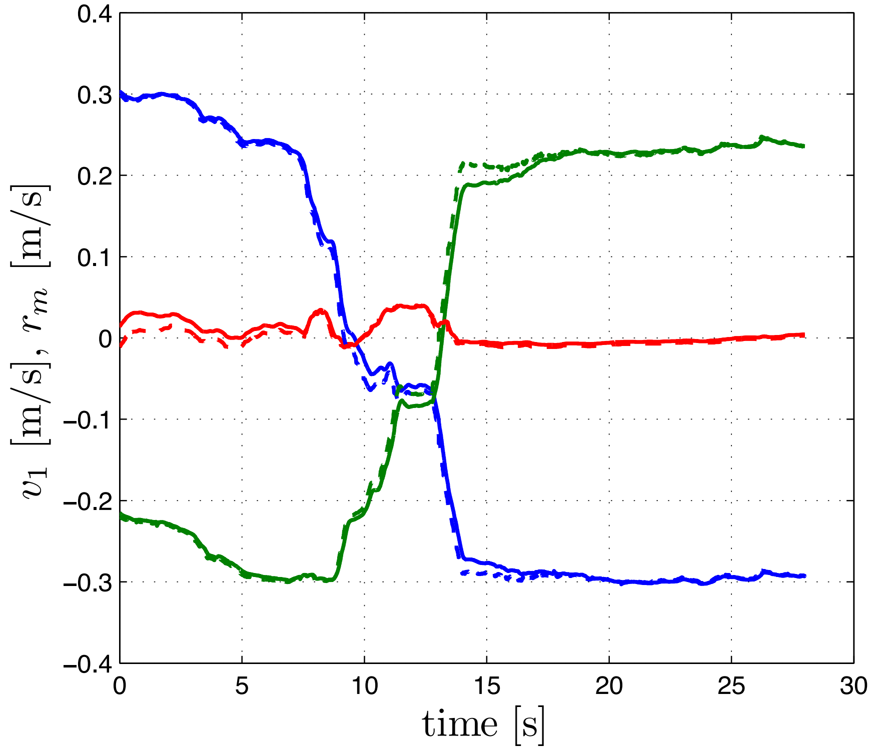
$$\dot{\tilde{\mathcal{H}}} = -\frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}} \mathcal{L} \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} - \frac{\partial^T \tilde{\mathcal{H}}}{\partial \tilde{p}_1} b_T \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}_1} \leq 0$$

- Must study the properties of the set $S = \{(\tilde{p}, x, t) \mid \dot{\tilde{H}} = 0\}$ (~ LaSalle)
- In this set, it is $\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} \in \ker \mathcal{L}$ and $\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}_1} = 0$
- Since $\ker \mathcal{L} = \mathbf{1}_{N_3}$, the set S is characterized by $\boxed{\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = 0}$
- From the dynamics -> $\dot{x} = -E^T \frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = 0$ (proof of Item 2)
- The condition $\frac{\partial \tilde{\mathcal{H}}}{\partial \tilde{p}} = 0$ implies $v - \mathbf{1}_{N_3} r_M = M(v - \mathbf{1}_{N_3} r_M) = \tilde{p} = 0$ and $\dot{\tilde{p}} = 0$ (proof of Item 1)

Velocity Synchronization

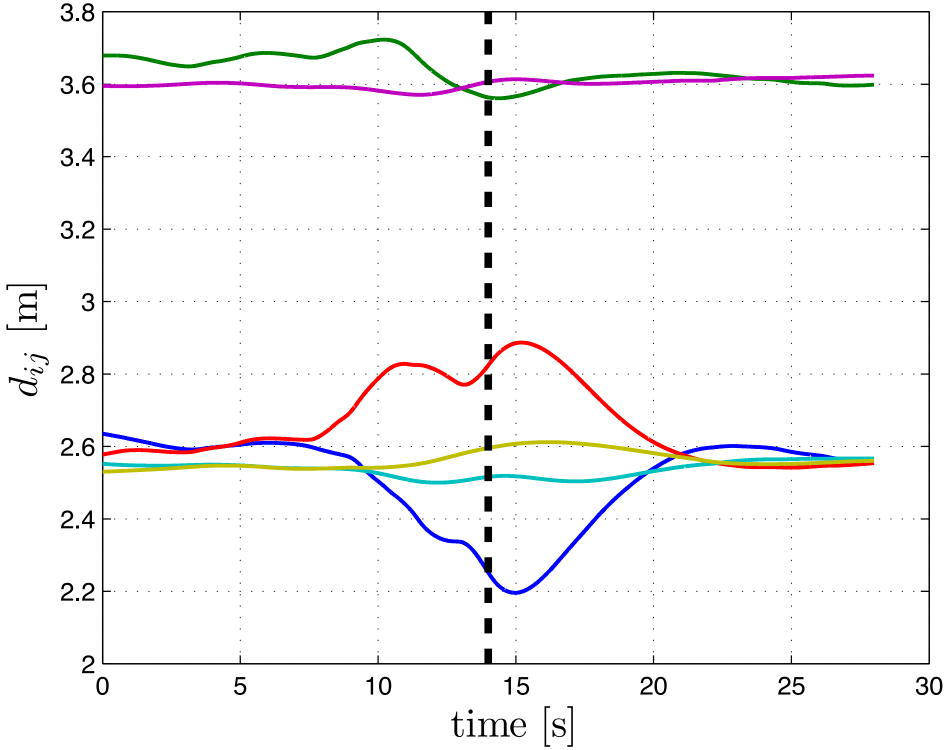


Master velocity commands r_M

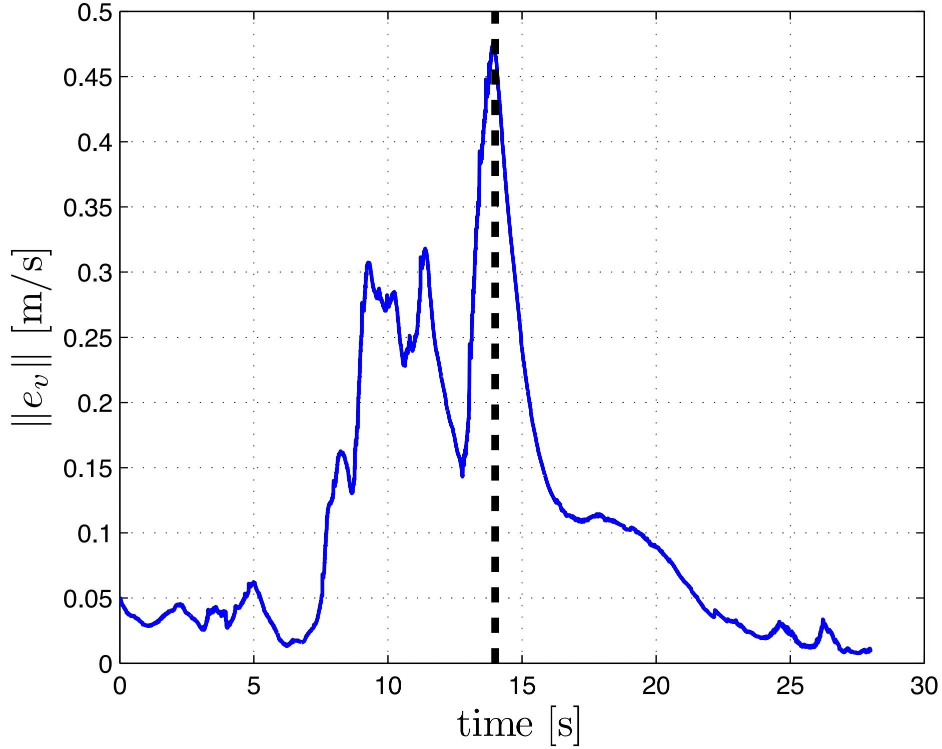


Leader vel. v_1 vs. r_M

Velocity Synchronization



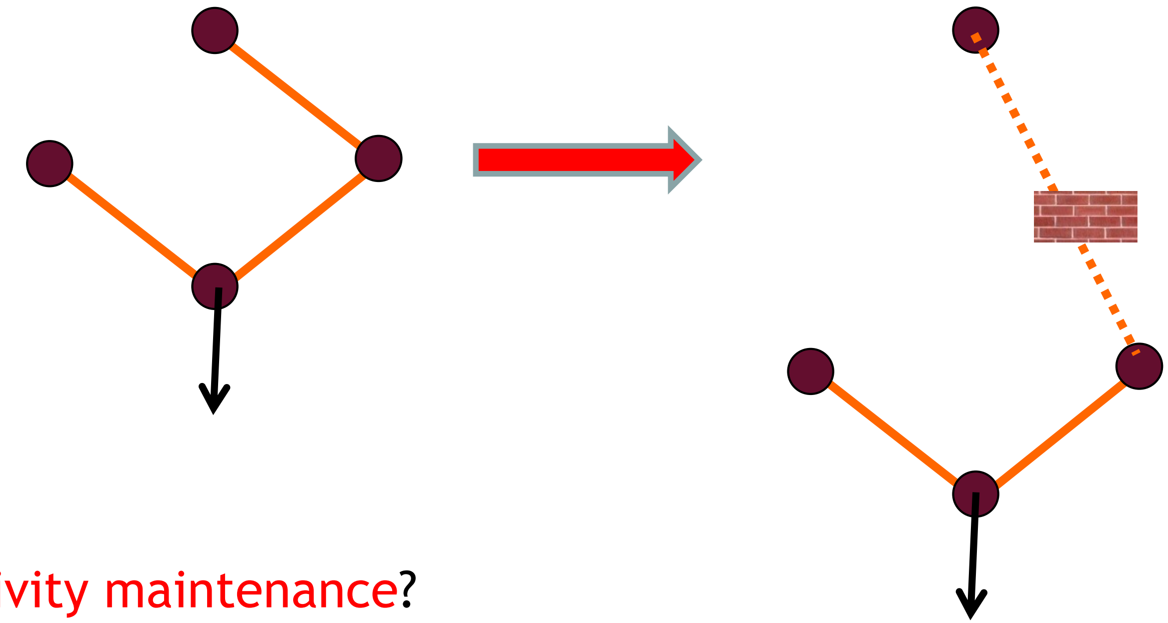
Interdistances



Norm of velocity synchronization error

$$\|e_v\| = \|v - \mathbf{1}_{N_3} r_M\|$$

Connectivity Maintenance

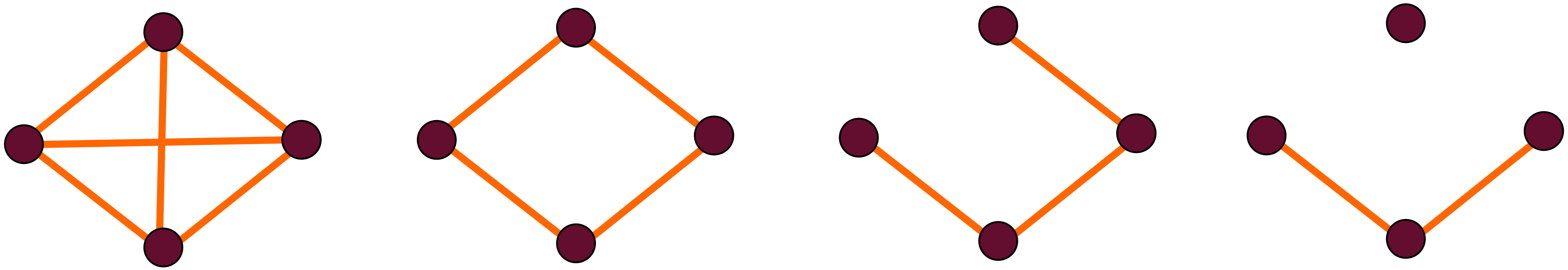


- What about connectivity maintenance?
- Can the graph $\mathcal{G}$ stay connected while still allowing **arbitrary split** and **join** as before?
- And...
- How to do it in a decentralized and stable/passive way?

Connectivity Maintenance

- Connected graph -> $\lambda_2 > 0$ (second smallest eigenvalue of the graph Laplacian L)

- λ_2 is a measure of the degree of connectivity in a graph
  - The larger its value, the "more connected" the graph



$\lambda_2 = 4$ $\quad$ $\lambda_2 = 2$ $\quad$ $\lambda_2 = 0.58$ $\quad$ $\lambda_2 = 0$

- However:
  - λ_2 is a global quantity $\longrightarrow$ against decentralization?
  - λ_2 does not vary smoothly over time $\longrightarrow$ cannot take "derivatives"

Connectivity Maintenance

- **As illustration**, it would be nice if one could have $\lambda_2 = \lambda_2(x)$ and then just implement some gradient-like controller $u = \dfrac{\partial \lambda_2}{\partial x}$

- This situation is actually possible: assume the weights of the Adjacency matrix are smooth functions of the state $A_{ij} = A_{ij}(x) \geq 0$ rather than $A_{ij} = \{0,\ 1\}$

- Then, the Laplacian itself becomes a smooth function of the state

$$\boxed{L = \Delta(x) - A(x) = L(x)}$$

- Let v_2 be the normalized eigenvector associated to λ_2

- By definition, it is $\lambda_2 = v_2^T L v_2$

- Then, $\boxed{\mathrm{d}\lambda_2 = \mathrm{d}v_2^T L v_2 + v_2^T \mathrm{d}L v_2 + v_2^T L \mathrm{d}v_2}$

Connectivity Maintenance

- How can we simplify $\mathrm{d}\lambda_2 = \mathrm{d}v_2^T L v_2 + v_2^T \mathrm{d}L v_2 + v_2^T L \mathrm{d}v_2$?

- Fact 1: since L is symmetric, it is $\mathrm{d}v_2^T L v_2 = v_2^T L \mathrm{d}v_2$

- Fact 2: $\mathrm{d}v_2^T L v_2 = \lambda_2 \mathrm{d}v_2^T v_2 = 0$ since v_2 is a normalized vector ($\|v_2\| = 1$)

- Then, $\boxed{\mathrm{d}\lambda_2 = v_2^T \mathrm{d}L v_2}$

- This implies that $\dfrac{\partial \lambda_2}{\partial x_i} = \displaystyle\sum_{(i,j)\in\mathcal{E}} \dfrac{\partial A_{ij}}{\partial x_i}(v_{2_i} - v_{2_j})^2$(follows from the definition of L)

- Note the nice "decentralized structure": **sum over the neighbors**

Connectivity Maintenance

- Now, we are just left with a proper design of the weights $A_{ij}(x)$

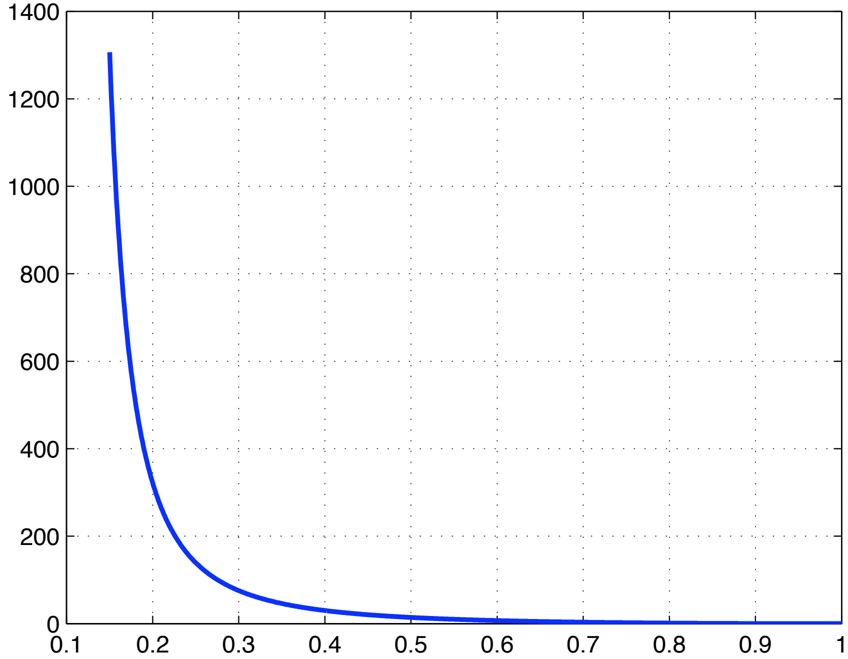
- The weights $A_{ij}(x)$ should possess the following features:
  - 1) they should be function of relative quantities, e.g., $A_{ij}(x_i - x_j)$
  - 2) they should smoothly vanish as a disconnection is approaching
    - for example, $A_{ij}(x_i - x_j) \to 0$ as $d_{ij} \to D$ for the max. range constraint

- One can exploit and extend this idea in order to embed in the weights A_{ij}
  - presence of physical limitations for interacting (e.g., occlusions, maximum range)
  - additional agent requirements which should be preferably met (e.g., keeping a desired interdistance)
  - additional agent requirements which must be necessarily met (avoiding collisions with obstacles and other agents)

- Everything achieved by the sole "maximization" of the unique scalar quantity $\lambda_2(x)$
  - **"physical" connectivity** + any **additional group requirement**

Connectivity Maintenance

- Possible control design: define a Connectivity Potential function $V^\lambda(\lambda_2) \geq 0$ which
  - vanishes for $\lambda_2 \rightarrow \lambda_2^{\max}$
  - grows unbounded for $\lambda_2 \rightarrow \lambda_2^{\min} < \lambda_2^{\max}$



- Its anti-gradient (connectivity force) w.r.t. the i-th agent position is

$$F_i^\lambda = -\frac{\partial V^\lambda(\lambda_2(x))}{\partial x_i} = -\frac{\partial V^\lambda(\lambda_2)}{\partial \lambda_2}\frac{\partial \lambda_2(x)}{\partial x_i}$$

- Next step: how to design the weights $A_{ij}(x)$? How to best take advantage of the flexibility of this framework?
- … and how to still cope with usual "**decentralization constraints**"

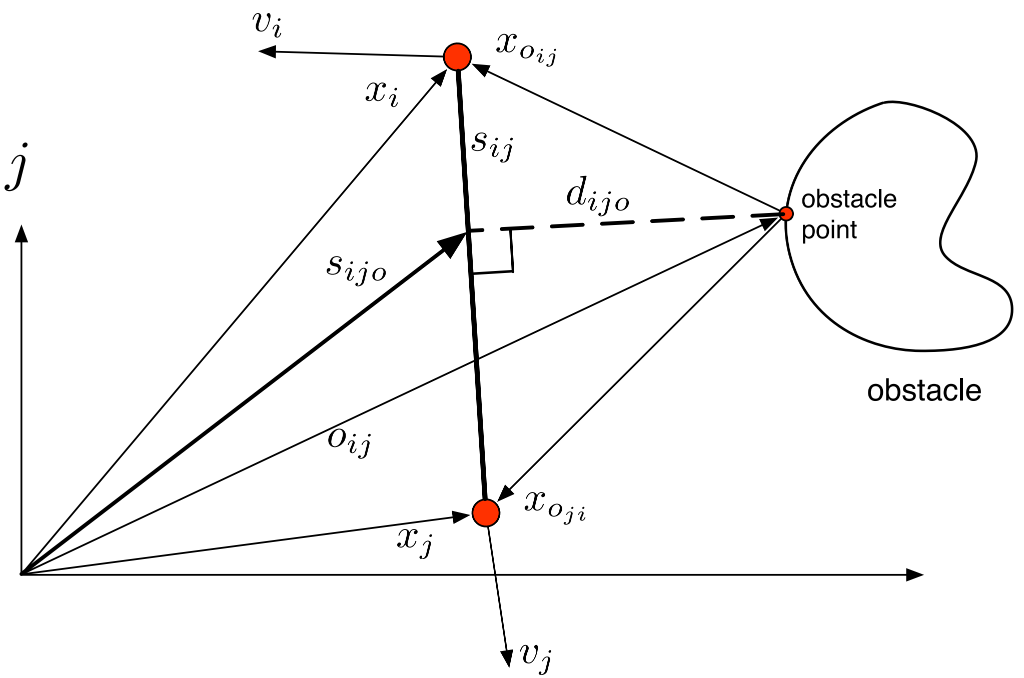
Connectivity Maintenance

- Let us see a possible choice for the various weight functions in $A_{ij} = \alpha_{ij}\beta_{ij}\gamma_{ij}$

- We define the set $\mathcal{S}_i = \{j|\ \gamma_{ij} > 0\}$ as the "sensing-neighbors" of agent i and the set $\mathcal{N}_i = \{j|\ A_{ij} > 0\}$ as the usual neighbors of agent i

- As for sensing/communication constraints, we consider (like before) maximum range and line-of-sight occlusion

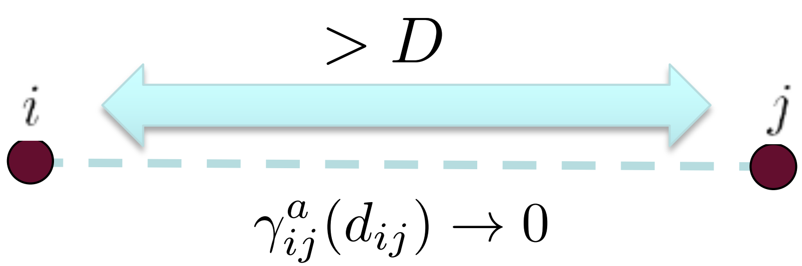
- We consider the following definitions:
  - s_{ij} is the segment joining agents i and j
  - o_{ij} is the closest obstacle point to s_{ij}
  - s_{ijo} is the closest point on s_{ij} to o_{ij}
  - d_{ijo} is the distance from s_{ij} to o_{ij}

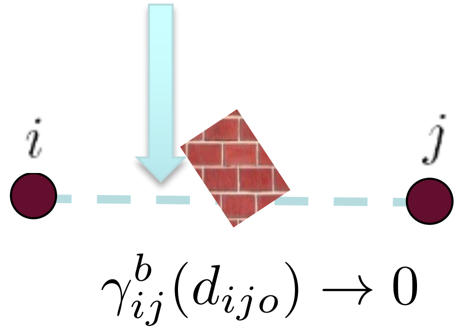


Connectivity Maintenance

- The term $\boxed{\gamma_{ij} \geq 0}$ accounts for "physical" limitations in the relative sensing/communication, it represents the **sensing/communication model**

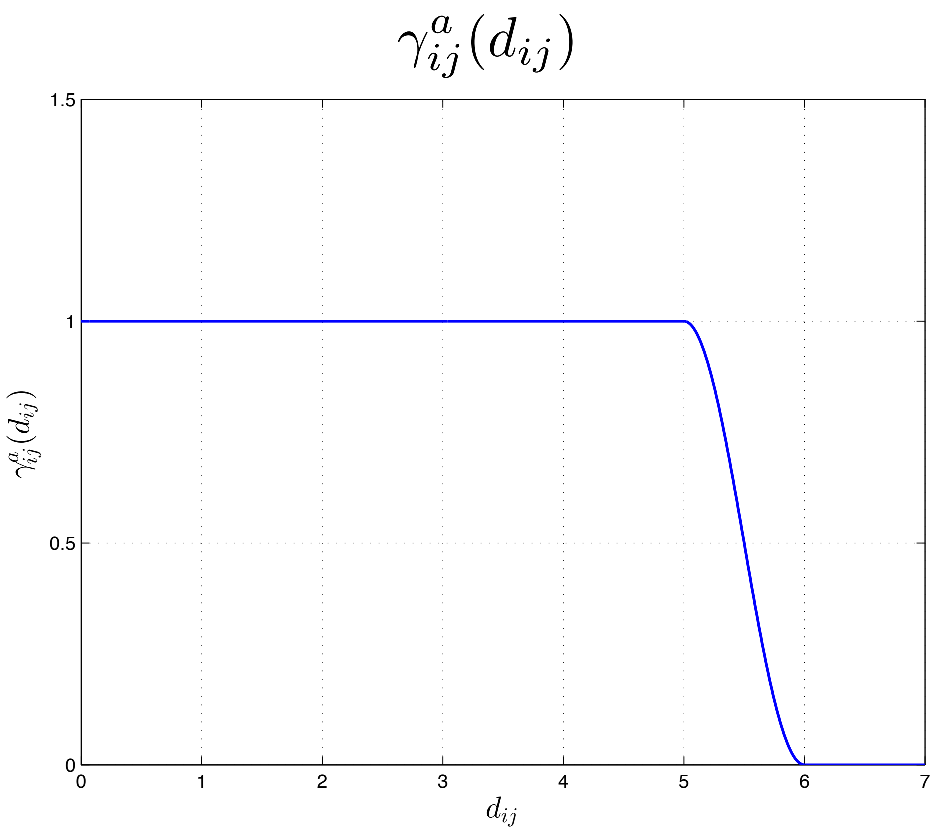
  - For instance, take $\gamma_{ij} = \gamma_{ij}^a(d_{ij})\gamma_{ij}^b(d_{ijo})$ where d_{ijo} is the distance from the segment joining agents i and j and the closest obstacle point o_{ij} and design
  - $\gamma_{ij}^a(d_{ij}) \to 0$ when exceeding the maximum range ($d_{ij} \to D$)
  - $\gamma_{ij}^b(d_{ijo}) \to 0$ when being occluded by an obstacle ($d_{ijo} \to 0$)

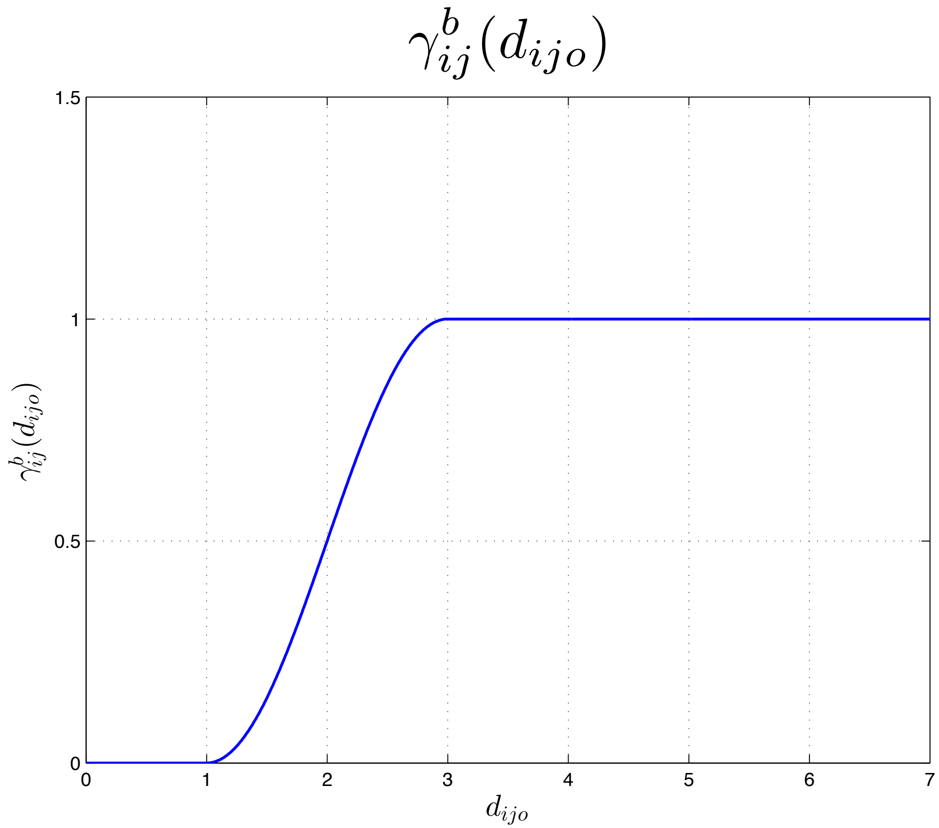




Connectivity Maintenance

- The weight $\gamma_{ij} = \gamma_{ij}^{a}(d_{ij})\gamma_{ij}^{b}(d_{ijo})$ is made of two terms
- $\gamma_{ij}^{a}(d_{ij}) \geq 0$ accounts for the maximum range constraint
- $\gamma_{ij}^{b}(d_{ijo}) \geq 0$ accounts for the minimum distance between line-of-sight and obstacles

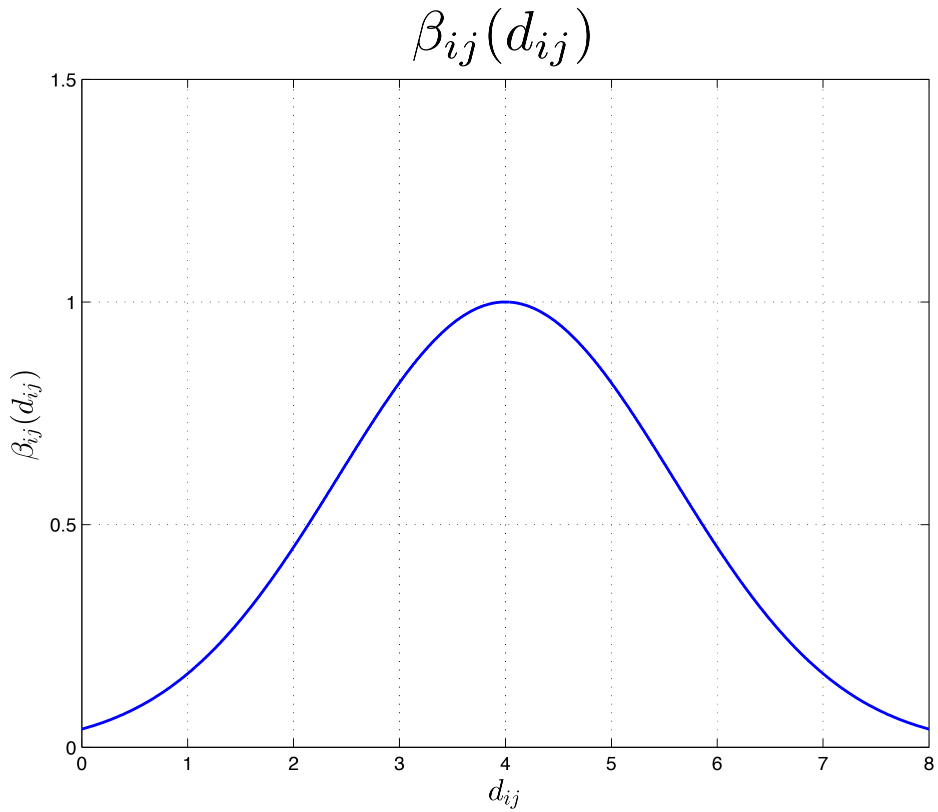




Connectivity Maintenance

- The term $\boxed{\beta_{ij} \geq 0}$ accounts for "**soft requirements**" that should be "preferably" realized by the agents (e.g., keep a desired distance)
  - For instance, $\beta_{ij}(d_{ij}) \to 0$ as $\|d_{ij} - d_0\| \to \infty$, and $\beta_{ij}(d_{ij})$ has a unique maximum at $d_{ij} = d_0$

$$A_{ij} = \alpha_{ij} \boxed{\beta_{ij}} \gamma_{ij}$$



Connectivity Maintenance

- The last term $\boxed{\alpha_{ij} \geq 0}$ accounts for "hard/mandatory" requirements that must be necessarily realized by the agents (e.g., avoid collisions)
  - As before, $\alpha_{ij}(d_{ij}) \to 0$ as $d_{ij} \to 0$ (i and j disconnect if they get too close)
  - **But also:** $\boxed{\alpha_{ik} \to 0, \ \forall k \in \mathcal{N}_i}$ (all the neighbors of i will disconnect!)
  - Approaching another agent will necessarily lead to a disconnected graph ($\lambda_2 \to 0$)

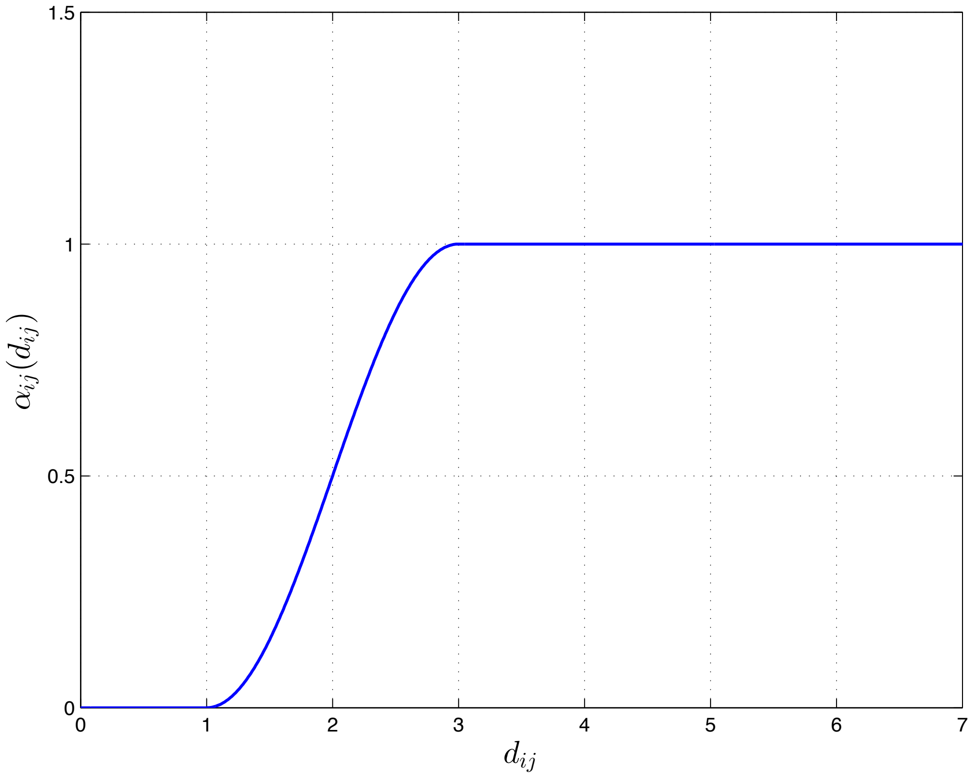
$$A_{ij} = \boxed{\alpha_{ij}}\beta_{ij}\gamma_{ij}$$

Connectivity Maintenance

- The weight $\alpha_{ij} \geq 0$ is made of a product of several terms $\alpha^*_{ij}(d_{ij}) \geq 0$

$$\alpha_{ij} = \left(\prod_{k \in \mathcal{S}_i} \alpha^*_{ik}\right) \cdot \left(\prod_{k \in \mathcal{S}_j / \{i\}} \alpha^*_{jk}\right) = \alpha_i \alpha_j$$

- The individual $\alpha^*_{ij}(d_{ij}) \to 0$ as $d_{ij} \to 0$ (a pair of agents $(i,\ j)$ gets too close)



Connectivity Maintenance

- However, we want to enforce that, if $d_{ij} \to 0$ for a particular pair $(i,\ j)$, the **whole graph approaches disconnection**. This is the reason for this form

$$\alpha_{ij} = \left(\prod_{k\in\mathcal{S}_i} \alpha^*_{ik}\right) \cdot \left(\prod_{k\in\mathcal{S}_j/\{i\}} \alpha^*_{jk}\right) = \alpha_i\alpha_j$$

- The term α_i in the product is constructed so that the very same term α_i will be present in all $\alpha_{ik},\ \forall k \in \mathcal{N}_i$ (for all neighbors of i)

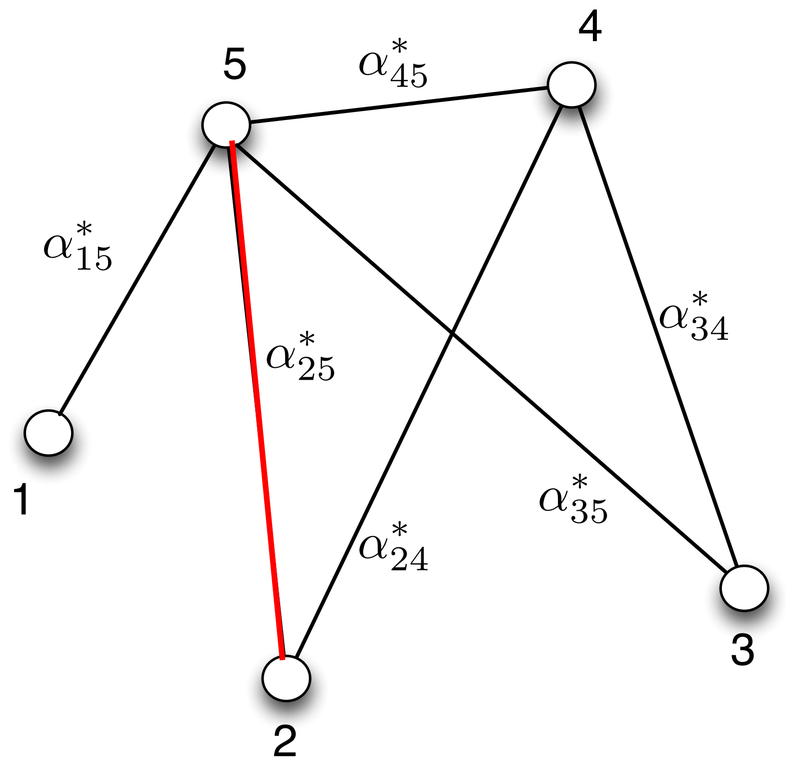
- Then, if agent i gets too close to some other agent, the entire $i-\mathrm{th}$ row of matrix A will vanish

- The term α_j is introduced for "symmetrizing" α_{ij} , i.e., to make sure that $\alpha_{ij} = \alpha_{ji}$

- The other weights are already symmetric by construction ($\beta_{ij} = \beta_{ji}$ and $\gamma_{ij} = \gamma_{ji}$)

- This ensures that, eventually, $A_{ij} = A_{ji}$ and $A = A^T$ (symmetric Adjacency matrix)

Connectivity Maintenance



- An example of the use of weights α_{ij}
- Consider the Graph in figure
- We have: $\alpha_{25} = (\alpha^*_{24}\alpha^*_{25})(\alpha^*_{51}\alpha^*_{53}\alpha^*_{54})$

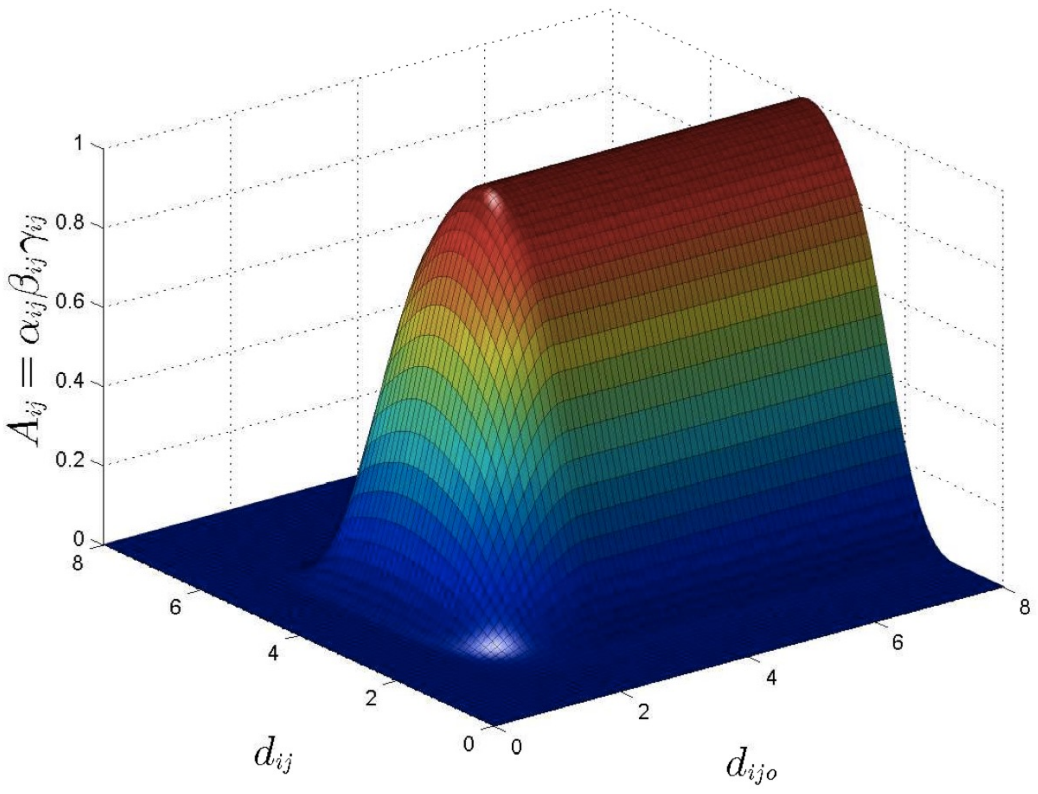
$$\alpha_{52} = (\alpha^*_{51}\alpha^*_{52}\alpha^*_{53}\alpha^*_{54})(\alpha^*_{24})$$

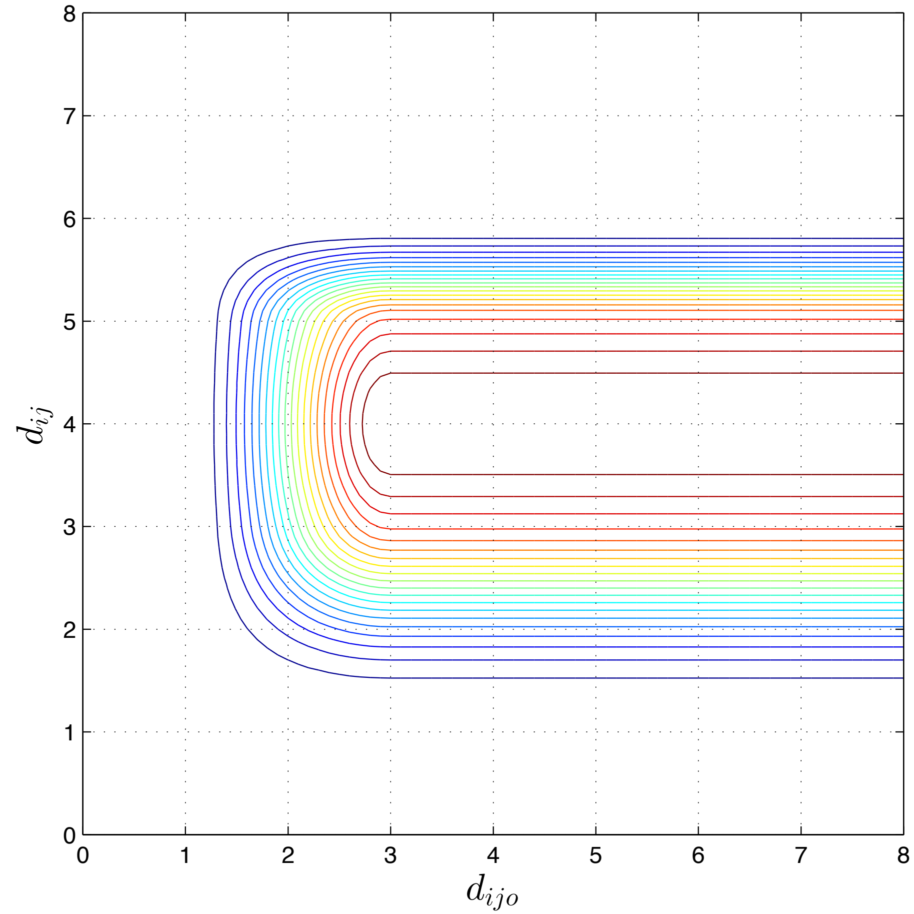
so that the "symmetry" condition $\alpha_{25} = \alpha_{52}$ holds

- Also, evaluating $\alpha_{24} = (\alpha^*_{24}\alpha^*_{25})(\alpha^*_{43}\alpha^*_{45})$ we note the common factor $(\alpha^*_{24}\alpha^*_{25})$
- This common factor makes the whole second row of A to vanish as agent 2 approaches any sensed agent
- We finally note that in $\alpha_{ij} = \left(\prod_{k\in\mathcal{S}_i} \alpha^*_{ik}\right)\cdot\left(\prod_{k\in\mathcal{S}_j/\{i\}} \alpha^*_{jk}\right) = \alpha_i\alpha_j$ the term α_j **does not depend on** x_i

Connectivity Maintenance

- Graphical representation of the total weight $A_{ij} = \alpha_{ij}\beta_{ij}\gamma_{ij}$ as a function of d_{ij} and d_{ijo}



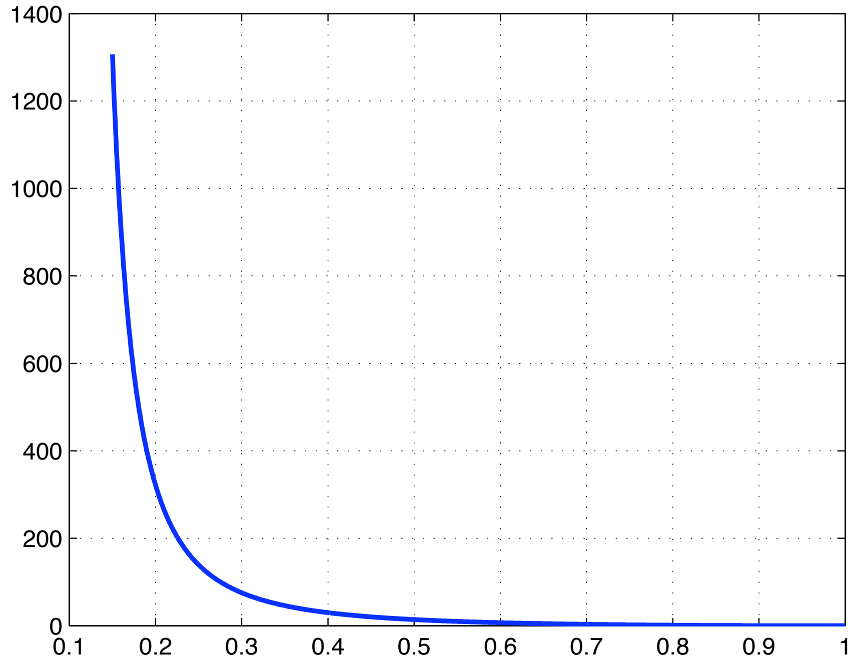


Connectivity Maintenance

- We have almost all the ingredients

- In view of the PHS form of the group dynamics, we still lack an Energy (Hamiltonian) function

- Let us then define a Connectivity Potential function $V^{\lambda}(\lambda_2) \geq 0$ which
  - vanishes for $\lambda_2 \to \lambda_2^{\max}$
  - grows unbounded for $\lambda_2 \to \lambda_2^{\min} < \lambda_2^{\max}$



- This will be the Storage function for our passivity arguments

- Its gradient (connectivity force) is

$$F_i^{\lambda}(x) = \frac{\partial V^{\lambda}(\lambda_2(x))}{\partial x}$$

Connectivity Maintenance

- Let $x_{\mathrm{R}} = (x_{12}^T \; \ldots \; x_{1N}^T \; x_{23}^T \; \ldots \; x_{2N}^T \; \ldots \; x_{N-1N}^T)^T \in \mathbb{R}^{\frac{3N(N-1)}{2}}$ be the edges and $x_{\mathrm{O}} \in \mathbb{R}^{3N(N-1)}$ collect all the $x_{o_{ij}} = x_i - o_{ij}$ (agent-obstacle relative positions)

- The Connectivity Force is then a function of the state $F_i^\lambda = \dfrac{\partial V^\lambda(\lambda_2(x_R, x_O))}{\partial x_i}$

- **What is its expression?** First of all, it is $F_i^\lambda = \dfrac{\partial V^\lambda(\lambda_2)}{\partial \lambda_2} \dfrac{\partial \lambda_2(x_R, x_O)}{\partial x_i}$ and we know that

$$\frac{\partial \lambda_2(x_R, x_O)}{\partial x_i} = \sum_{j \in \mathcal{N}_i} \frac{\partial A_{ij}(x_R, x_O)}{\partial x_i} (v_{2_i} - v_{2_j})^2$$

- The weights $A_{ij} = \alpha_{ij}(x_R)\beta_{ij}(x_R)\gamma_{ij}(x_R, x_O)$. It is then sufficient to use the product rule for taking derivatives

- And since the weights are functions of relative quantities, we can **finally obtain**

$$\boxed{\frac{\partial \lambda_2(x_R, x_O)}{\partial x_i} = \sum_{j \in \mathcal{N}_i} \left(\frac{\partial A_{ij}}{\partial x_{ij}} + \frac{\partial A_{ij}}{\partial x_{ijo}} \right) (v_{2_i} - v_{2_j})^2}$$

Connectivity Maintenance

- What about decentralization of $F_i^\lambda = \frac{\partial V^\lambda(\lambda_2)}{\partial \lambda_2} \sum_{j \in \mathcal{N}_i} \left(\frac{\partial A_{ij}}{\partial x_{ij}} + \frac{\partial A_{ij}}{\partial x_{ijo}} \right) (v_{2_i} - v_{2_j})^2$?
- Evaluation of $\frac{\partial \gamma_{ij}}{\partial x_{ij}}$ and $\frac{\partial \gamma_{ij}}{\partial x_{ijo}}$ requires knowledge of x_i, x_j and o_{ij}
- Similarly for $\frac{\partial \beta_{ij}}{\partial x_{ij}}$
- The term $\alpha_{ij} = \alpha_i \alpha_j$ has two components: evaluation of $\frac{\partial \alpha_i}{\partial x_{ij}}$ requires again only x_i and x_j
- As for $\frac{\partial \alpha_j}{\partial x_{ij}}$, we noted that α_j does not depend on x_i , therefore $\boxed{\frac{\partial \alpha_j}{\partial x_{ij}} \equiv 0}$
- Finally, the quantity $\alpha_j = \prod_{k \in \mathcal{S}_j / \{i\}} \alpha^*_{jk}$ can be communicated as a **single scalar quantity** from agent j to agent i
- Therefore, only local and 1-hop information so far

Connectivity Maintenance

- However, we still need knowledge of λ_2, v_{2_i} and v_{2_j}, $j \in \mathcal{N}_i$ for fully implementing

$$F_i^\lambda = \frac{\partial V^\lambda(\lambda_2)}{\partial \lambda_2} \sum_{j \in \mathcal{N}_i} \left(\frac{\partial A_{ij}}{\partial x_{ij}} + \frac{\partial A_{ij}}{\partial x_{ijo}} \right) (v_{2_i} - v_{2_j})^2$$

- Knowledge of these quantities could obtained in a centralized way (need to know the full Laplacian L)

- Alternatively, one can resort to a decentralized estimation yielding suitable (estimated) values $\hat{\lambda}_2$, $\hat{v}_{2_i}$ and $\hat{v}_{2_j}$, $j \in \mathcal{N}_i$

- This allows for a fully decentralized implementation of $\hat{F}_i^\lambda$ (an estimation of the real F_i^λ)

- More details later on this estimation strategy

Connectivity Maintenance

- As for the agent dynamics, we consider the usual

$$\begin{cases} \dot{p}_i = F_i^\lambda + F_i^e - B_i M_i^{-1} p_i \\ v_i = \dfrac{\partial \mathcal{K}_i}{\partial p_i} = M_i^{-1} p_i \end{cases}$$

- Putting all together, we can finally model the group dynamics as a **PHS**

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x}_R \\ \dot{x}_O \end{pmatrix} = \left[\begin{pmatrix} 0 & E & -\mathbb{I} \\ -E^T & 0 & 0 \\ \mathbb{I}^T & 0 & 0 \end{pmatrix} - \begin{pmatrix} B & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right] \nabla H + G \begin{pmatrix} F^e \\ v_o \end{pmatrix} \\ \begin{pmatrix} v \\ F^o \end{pmatrix} = G^T \nabla H \end{cases} .$$

with $H(p, x_R, x_O) = \sum_{i=1}^{N} \mathcal{K}_i(p_i) + V^\lambda(x_R, x_O) \geq 0$, $G = \begin{pmatrix} I_N \otimes I_3 & 0 & 0 \\ 0 & 0 & -\mathbb{I} \end{pmatrix}^T$

$\mathbb{I} = I_N \otimes \mathbf{1}_{N-1}^T \otimes I_3$, and $F^e = (F_1^{eT} \ \ldots \ F_N^{eT})^T \in \mathbb{R}^{3N}$ being the external forces on the agent group

- The pair (F^o, v_o) is associated to the "obstacle motion" (passive system)

Connectivity Maintenance

- The group (PHS system) should be passive w.r.t. its power ports

$$\dot{H} = \nabla^T H \begin{pmatrix} \dot{p} \\ \dot{x}_R \\ \dot{x}_O \end{pmatrix} = -\frac{\partial H}{\partial p} B \frac{\partial H}{\partial p} + \nabla^T H G \begin{pmatrix} F^{\mathrm{e}} \\ v_o \end{pmatrix} \leq v^T F^{\mathrm{e}} + v_o^T F^o.$$

- Still, two sources of possible non-passive behaviors can affect the system

- **First:** possible positive jumps in $V^\lambda(\lambda_2)$ because of join decisions (as before)

- **Second:** estimation errors in evaluating $\hat{F}_i^\lambda$ (in place of the real F_i^λ)

- Solution to first issue: no jump is possible for $V^\lambda(\lambda_2(t))$ in this new framework

- Reason: the entries of the Adjacency matrix $A_{ij}(x_R(t),\, x_O(t))$ will always vary as smooth functions of the state
  - No discontinuities in $A_{ij}(t)$ imply no discontinuities in $\lambda_2(t)$ which, in turn, imply no discontinuities in $V(\lambda_2(t))$

Connectivity Maintenance

- What about estimation errors in $\hat{F}_i^\lambda$? These can be a source of non-passivity

- **Solution: use Tanks!**
  - Store dissipated energy, and use this energy for implementing $\hat{F}_i^\lambda$

- New agent dynamics augmented with the Tank element

$$\begin{cases} \dot{p}_i = F_i^e - w_i x_{t_i} - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} = s_i \frac{1}{x_{t_i}} D_i + w_i^T v_i \\ y_i = \left(v_i^T \ x_{t_i}\right)^T \end{cases} .$$

- The parameter $s_i = \begin{cases} 0, & \text{if } T_i \geq T_{\max} \\ 1, & \text{if } T_i < T_{\max} \end{cases}$ prevents excessive storage in the Tank

- Force $\hat{F}_i^\lambda$ is then implemented by setting $w_i = -\varsigma_i \dfrac{\hat{F}_i^\lambda}{x_{t_i}}, \quad \varsigma_i \in \{0, 1\}$

Connectivity Maintenance

- The additional parameter $\varsigma_i = \begin{cases} 0, & \text{if } T_i < T_{\min} \\ 1, & \text{if } T_i \geq T_{\min} \end{cases}$ enables/disables the implementation of $\hat{F}_i^\lambda$ when close to Tank depletion

- Considering the Tank dynamics, the agent group becomes

$$\begin{cases} \begin{pmatrix} \dot{p} \\ \dot{x}_R \\ \dot{x}_O \\ \dot{x}_t \end{pmatrix} = \left[\begin{pmatrix} 0 & E & -\mathbb{I} & \Upsilon \\ -E^T & 0 & 0 & 0 \\ \mathbb{I}^T & 0 & 0 & 0 \\ -\Upsilon^T & 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} B & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -SPB & 0 & 0 & 0 \end{pmatrix} \right] \nabla \mathcal{H} + G \begin{pmatrix} F^e \\ v_o \end{pmatrix} \\ \begin{pmatrix} v \\ F^o \end{pmatrix} = G^T \nabla \mathcal{H} \end{cases}$$

with $\mathcal{H}(p, x_R, x_O, x_t) = \sum_{i=1}^{N} (\mathcal{K}_i(p_i) + T_i(x_{t_i})) + V^\lambda(x_R, x_O)$ new Hamiltonian and

$P = diag(\frac{1}{x_{t_i}} p_i^T M_i^{-T}) \in \mathbb{R}^{N \times 3N}$, $S = diag(s_i) \in \mathbb{R}^{N \times N}$ and matrix

$\Upsilon = diag(-w_i) \in \mathbb{R}^{3N \times N}$ representing the Energy exchange between Tanks and agents

Connectivity Maintenance

- The group dynamics is passive (**proof as exercise**) $\dot{\mathcal{H}} \leq v^T F^{\mathrm{e}} + v_o^T F^o$

- The Tank machinery exploited for implementing $\hat{F}_i^\lambda$ ensures passivity of the group but it does not automatically guarantee Connectivity Maintenance

- The Connectivity Force $\hat{F}_i^\lambda$ could not be implemented because the Tank is depleted
  - Because of too poor estimation of F_i^λ over time

- This could happen in general, but it does not happen in our particular situation

- **Fact 1:** when exploiting a Tank Energy for implementing the port behavior of a passive system (as (F_i^λ, v_i) in our case), it can be shown that the Tank will never deplete (provided a correct initialization of $T(x_{t_i}(t_0))$)

- **Fact 2:** the estimation strategy used for $\hat{F}_i^\lambda$ is guaranteed to have a bounded error (with tunable accuracy)

Connectivity Maintenance

- Therefore, one can set up the system (e.g., choose gains) so that any residual non-passive effect due to estimation errors is dominated by the storing of the agent dissipation D_i

$$\begin{cases} \dot{p}_i = F_i^e - w_i x_{t_i} - B_i M_i^{-1} p_i \\ \dot{x}_{t_i} = \boxed{s_i \frac{1}{x_{t_i}} D_i} + w_i^T v_i \\ y_i = \left(v_i^T \; x_{t_i}\right)^T \end{cases} .$$

- Then, Tanks will never deplete over time!

- Let us have a look at the estimation strategy for $\hat{\lambda}_2$, $\hat{v}_{2_i}$ and $\hat{v}_{2_j}$, $j \in \mathcal{N}_i$

- The idea is to estimate (in a decentralized way) the eigenvector v_2

- This, in turn, allows to also estimate λ_2

- Recall that $v_2 \in \mathbb{R}^N$. Every agent maintains an estimation of its component v_{2_i}

Connectivity Maintenance

- Let $\hat{v}_2$ be the current estimate of the eigenvector v_2

- The estimation algorithm is a continuous-time version of the Power Iteration Procedure for computing eigenvectors and eigenvalues of a matrix

- It consists of **three steps**:

- 1) Deflation $\dot{\hat{v}}_2 = -\frac{k_1}{N}\mathbf{1}\mathbf{1}^T\hat{v}_2$ for removing the components spanned by $v_1 = \mathbf{1}$

- 2) Direction update $\dot{\hat{v}}_2 = -k_2 L\hat{v}_2$ for moving towards v_2

- 3) Renormalization $\dot{\hat{v}}_2 = -k_3\left(\frac{\hat{v}_2^T\hat{v}_2}{N} - 1\right)\hat{v}_2$ from staying away from the null-vector

- Altogether: $\boxed{\dot{\hat{v}}_2 = -\frac{k_1}{N}\mathbf{1}\mathbf{1}^T\hat{v}_2 - k_2 L\hat{v}_2 - k_3\left(\frac{\hat{v}_2^T\hat{v}_2}{N} - 1\right)\hat{v}_2}$

- An estimate for the connectivity eigenvalues is then obtained as $\hat{\lambda}_2 = \frac{1}{\|\hat{v}\|^2}\hat{v}_2^T L\hat{v}_2$

Connectivity Maintenance

- Is $\dot{\hat{v}}_2 = -\frac{k_1}{N}\mathbf{1}\mathbf{1}^T\hat{v}_2 - k_2 L\hat{v}_2 - k_3\left(\frac{\hat{v}_2^T\hat{v}_2}{N} - 1\right)\hat{v}_2$ decentralized?

- **<u>Almost:</u> everything is decentralized** apart from
  - the average $\frac{\mathbf{1}^T\hat{v}_2}{N}$
  - the average norm $\frac{\hat{v}_2^T\hat{v}_2}{N}$

- These last two quantities can be (themselves) estimated in a decentralized way by making use of the PI-ACE estimator (<u>**proportional/integral-Average Consensus Estimator**</u>)

$$\begin{cases} \dot{z}^i &= \gamma(\alpha^i - z^i) - K_P \sum_{j\in\mathcal{N}_i}(z^i - z^j) + K_I \sum_{j\in\mathcal{N}_i}(w^i - w^j) \\ \dot{w}^i &= -K_I \sum_{j\in\mathcal{N}_i}(z^i - z^j) \end{cases}$$

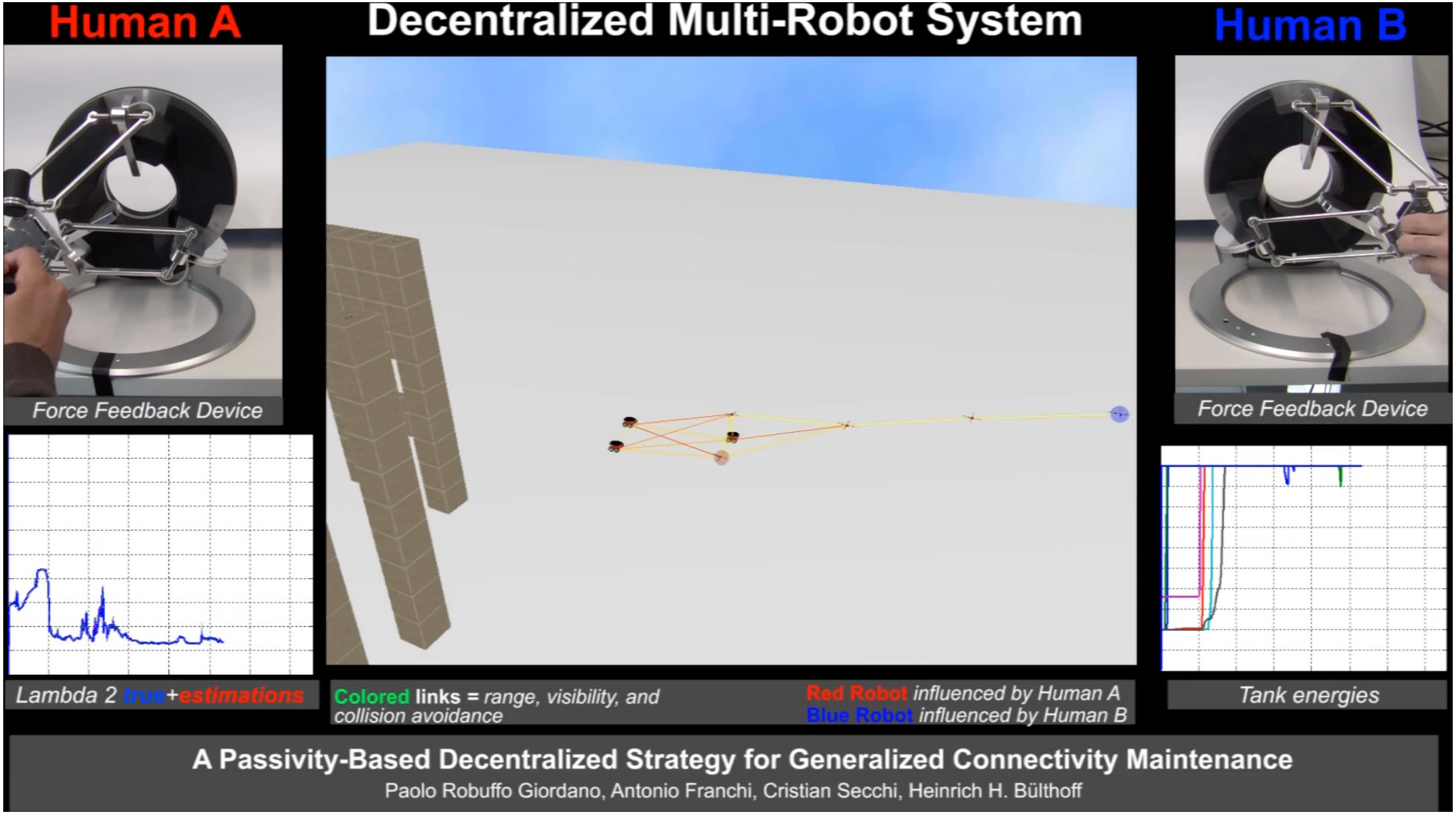
- The quantities $(z^i,\, w^i)$ are the PI-ACE states, and $\alpha_i = \{\hat{v}_{2_i},\, \hat{v}_{2_i}^2\}$are the external signals of which a moving average is taken (in a decentralized way)

Connectivity Maintenance

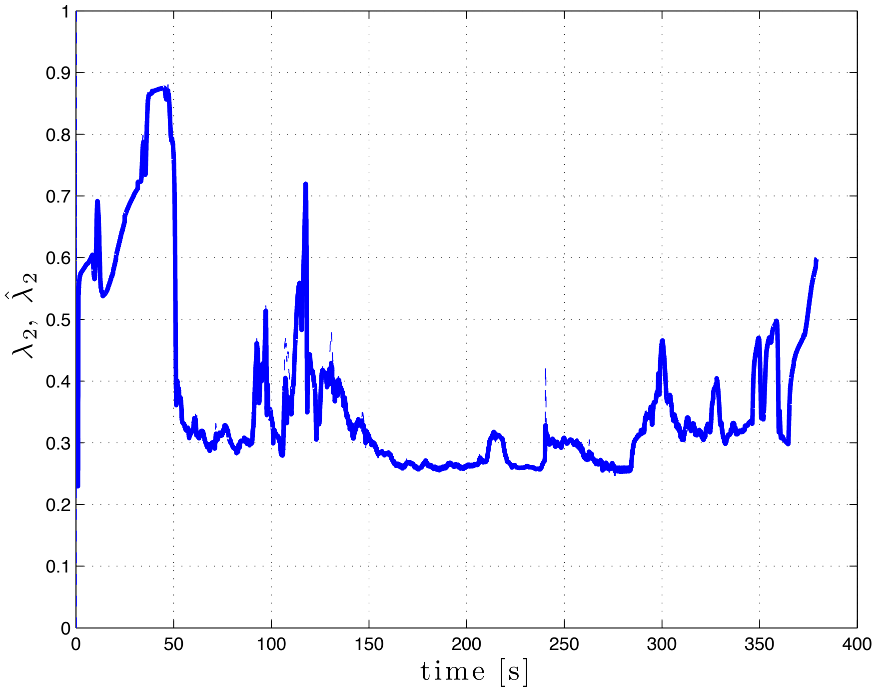
- Let us then see some results of this Connectivity Maintenance algorithm
- **Summarizing:** the single scalar quantity λ_2 encodes
  - physical connectivity (max. range, line-of-sight occlusion)
  - extra "soft-requirements" (keep a desired interdistance)
  - extra "hard-requirements" (avoid collisions with obstacles and agents)
  - still, possibility to split/join at anytime as long as the graph $\mathcal{G}$ stays connected
  - everything decentralized
  - everything passive (in **PHS** form, by making use of the Tank machinery)
- In the next simulations/videos, the usual group of N quadrotor UAVs
- **Two of them** are also commanded by two human operators
- The whole group must keep connectivity (as defined before)

Connectivity Maintenance

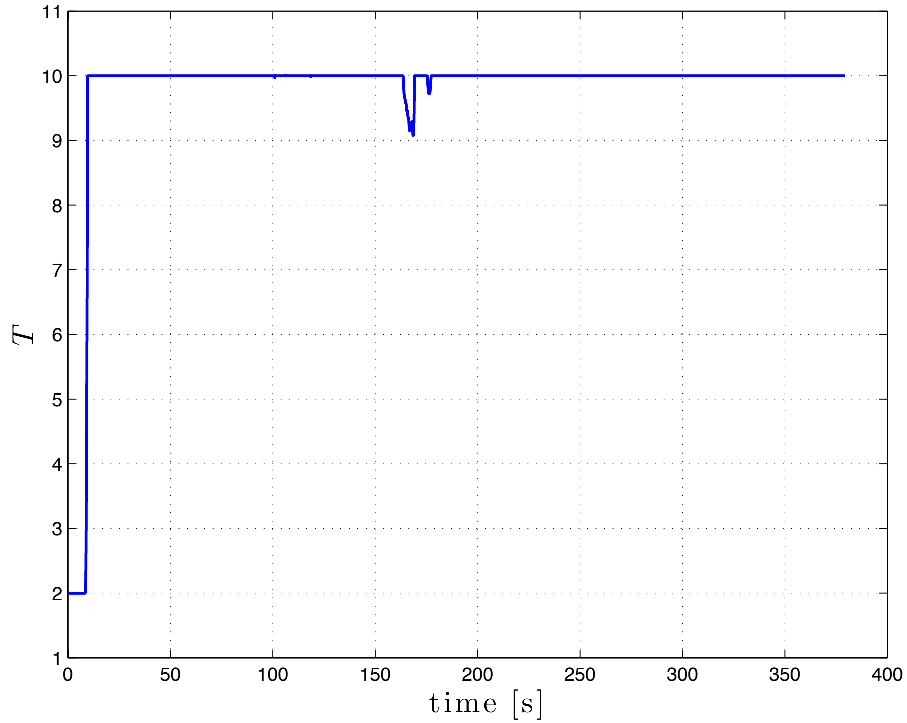
- Simulations with $N = 8$ robots (quadrotor UAVs and ground robots)



Connectivity Maintenance



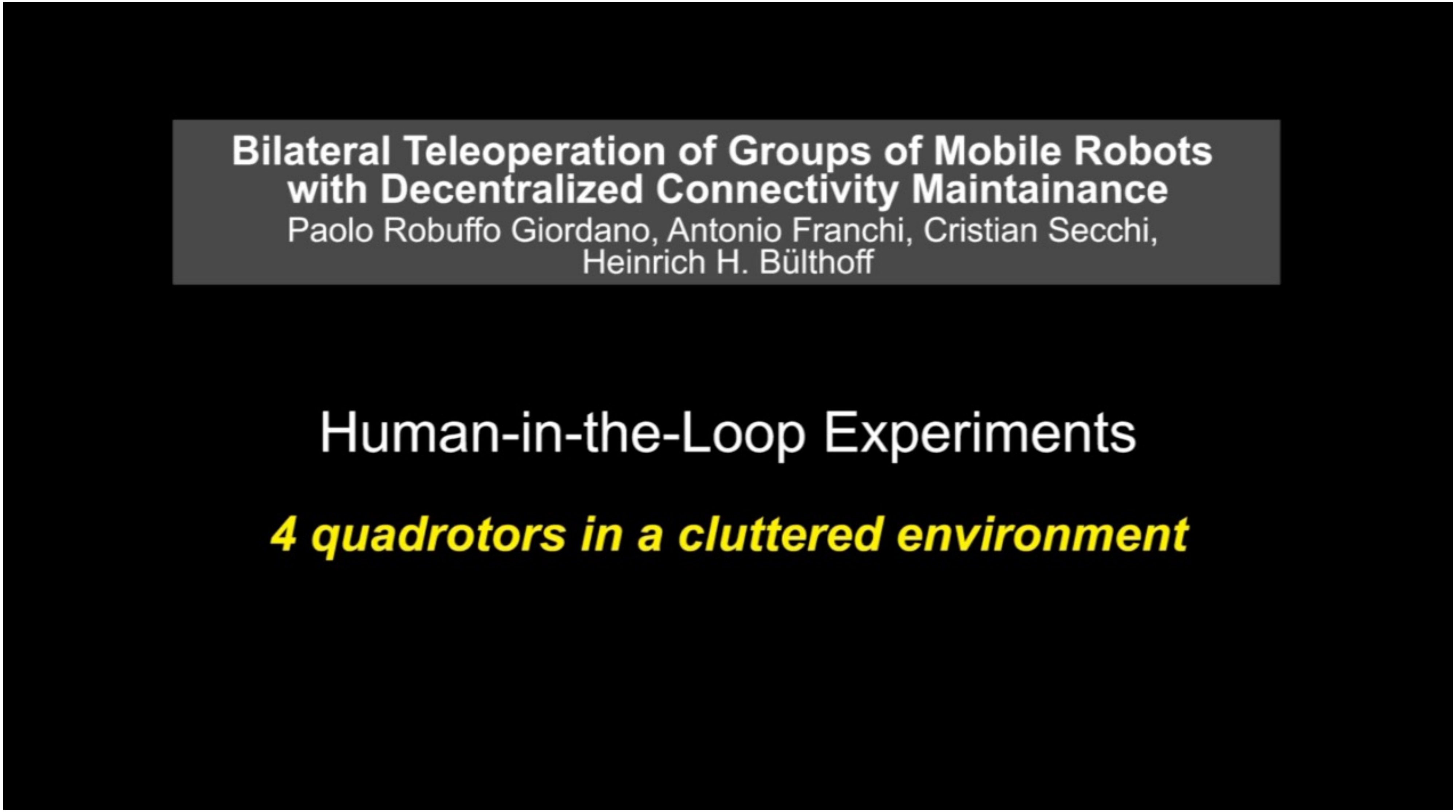
Real λ_2 (solid) vs. estimated $\hat{\lambda}_2^i$ (dashed)



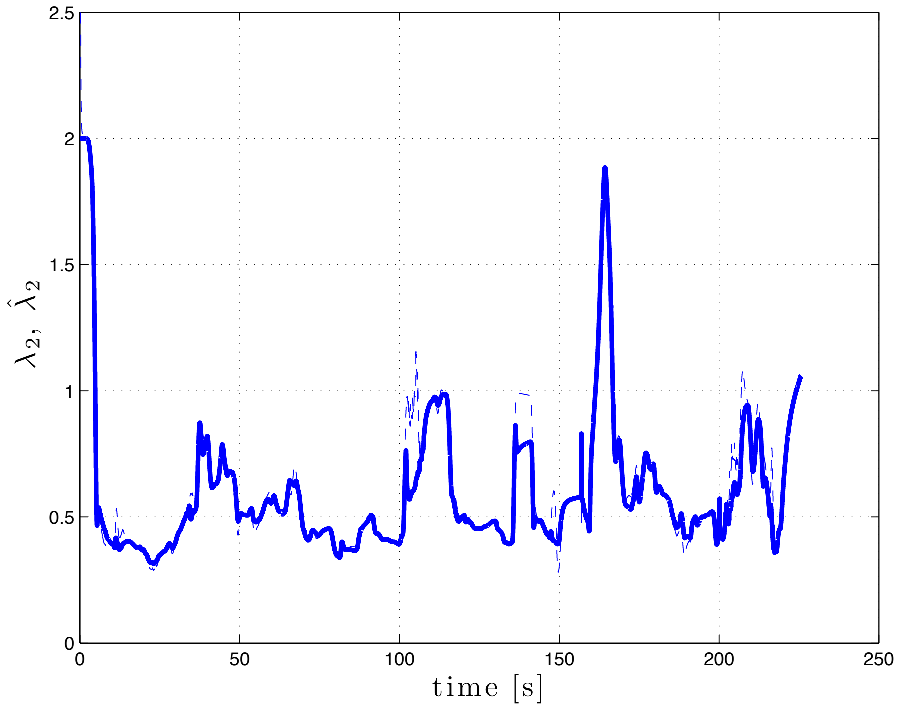
Tank energies $T(x_t)$

Connectivity Maintenance

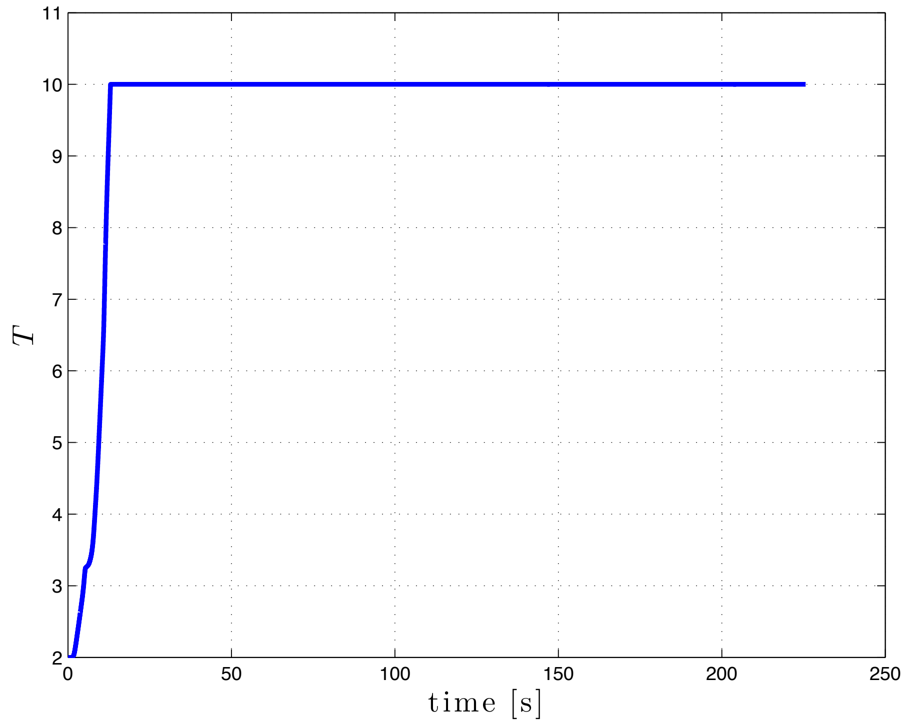
- Experiments with $N = 4$ quadrotor UAVs



Connectivity Maintenance



Real λ_2 (solid) vs. estimated $\hat{\lambda}_2^i$ (dashed)



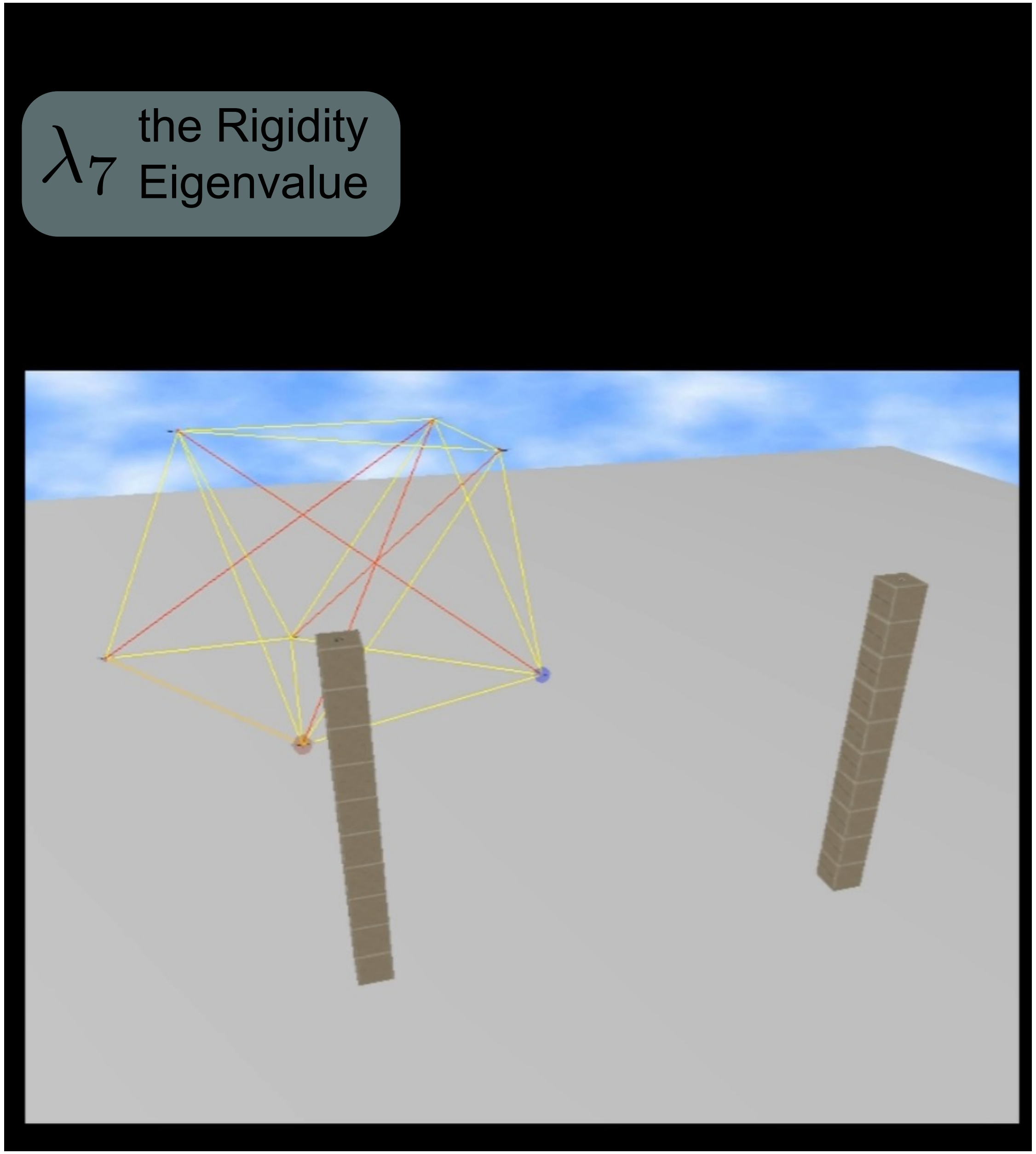
Tank energies $T(x_t)$

Rigidity Maintenance

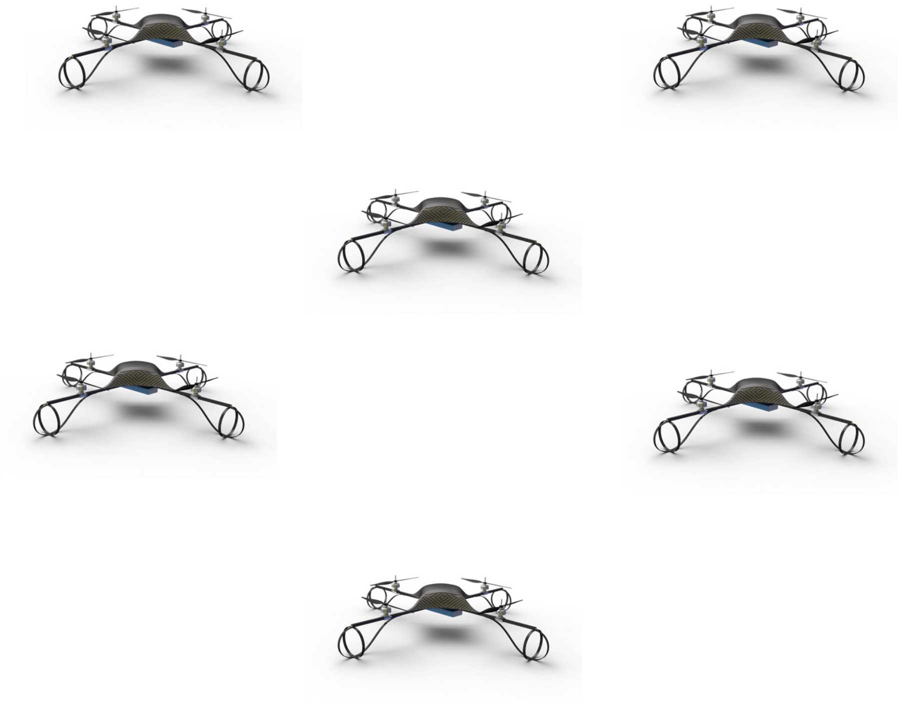
- One can also define a "Rigidity Eigenvalue" λ_7 and apply the same machinery

- rigidity maintenance with the same constraints and requirements as before

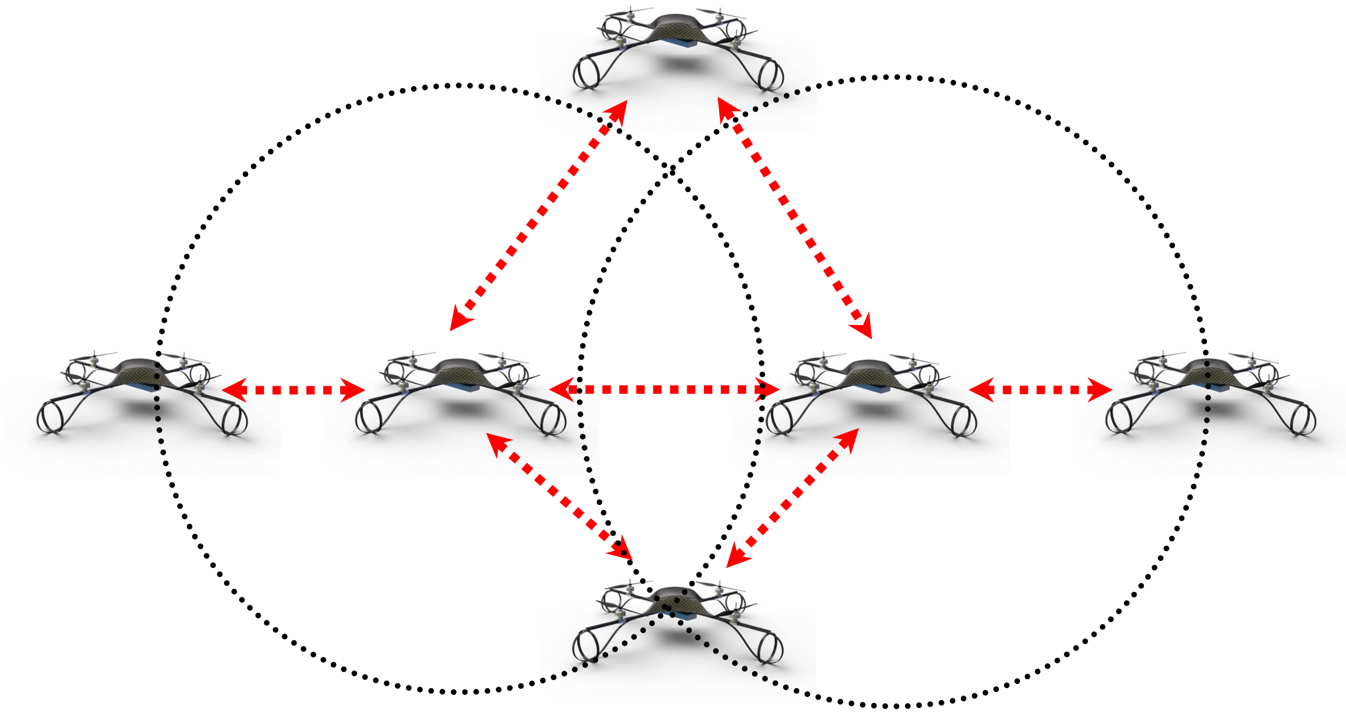
- Still flexibility in the graph topology $\lambda_7 > 0$



What is rigidity?



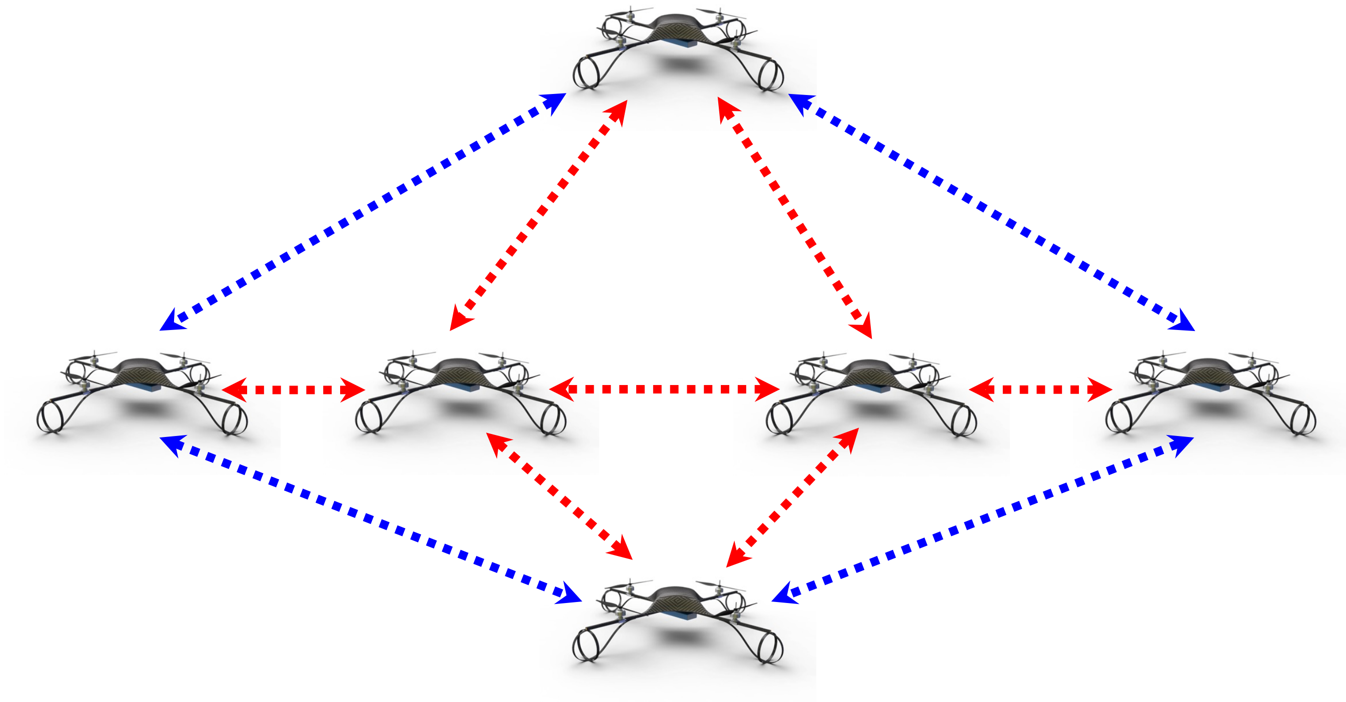
What is rigidity?



Can the desired formation be maintained using only the available distance measurements?

No!

What is rigidity?



A minimum number of distance measurements are required to uniquely determine the desired formation!

Graph Rigidity

Infinitesimal Rigidity (Lecture 2, Graph Theory)

- Infinitesimal rigidity: study the flexibility of a framework under instantaneous motions of its nodes

- Assume a smooth time dependence $p = p(t)$: what are the instantaneous motions of $p(t)$ which preserve the constraints $g_{\mathcal{G}}(p(t)) = const$?

- $g_{\mathcal{G}}(p(t)) = const \implies \dot{g}_{\mathcal{G}}(p(t)) = 0$ and using the chain rule

$$\boxed{\dot{g}_{\mathcal{G}}(p(t)) = 0 \implies \frac{\partial g_{\mathcal{G}}(p)}{\partial p}\dot{p} = R_{\mathcal{G}}(p)\dot{p} = 0}$$

- Matrix $\boxed{R_{\mathcal{G}}(p) \in \mathbb{R}^{|\mathcal{E}| \times Nd}}$ is known as the rigidity matrix

- The infinitesimal motions consistent with the constraints are then $\dot{p} \in \ker(R_{\mathcal{G}}(p))$

- A framework is infinitesimally rigid if $\ker(R_{\mathcal{G}}(p)) = \ker(R_K(p))$ or, equivalently, $\mathrm{rank}(R_{\mathcal{G}}(p)) = \mathrm{rank}(R_K(p))$
- Usual definition involving the complete graph K_N

Infinitesimal Rigidity (Lecture 2, Graph Theory)

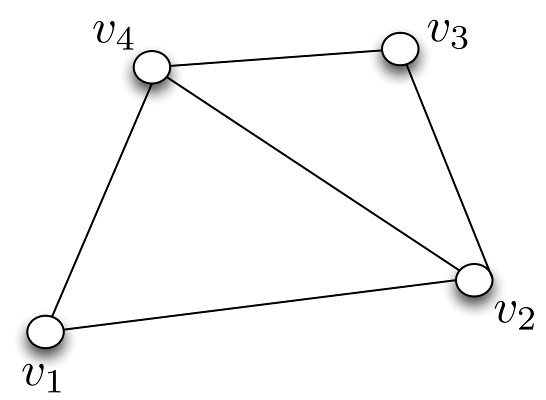
- The Rigidity matrix is a fundamental tool for control and estimation purposes
  - It establishes a link between agent motion and constraint variations
  - Its null-space $\ker(R_{\mathcal{G}}(p))$ describes all the motions preserving the constraints
  - Rigidity of a framework is equivalent to a rank condition on $R_{\mathcal{G}}(p)$. This allows to exploit spectral tools (e.g., eigenvalues, singular values) for checking or enforcing rigidity)

- The rank condition allows to also determine the minimum number of edges in a graph $\mathcal{G}$ for being rigid

- Let $\mathrm{rank}(R_{K_N}(p)) = r < Nd$. A framework is rigid if $\mathrm{rank}(R_{\mathcal{G}}(p)) = \mathrm{rank}(R_{K_N}(p))$

- Since $R_{\mathcal{G}}(p) \in \mathbb{R}^{|\mathcal{E}| \times Nd}$, this implies presence of at least $|\mathcal{E}| = r$ in the edge set of $\mathcal{G}$
  - However, not any collection of $|\mathcal{E}| = r$ edges would be good ! One needs the "right ones"

Infinitesimal Rigidity (Lecture 2, Graph Theory)



- Let us consider this graph

- What is the associated rigidity matrix ?

- Start with the constraint function $g(p) = \begin{bmatrix} \|p_1 - p_2\|^2 \\ \|p_1 - p_4\|^2 \\ \|p_2 - p_3\|^2 \\ \|p_2 - p_4\|^2 \\ \|p_3 - p_4\|^2 \end{bmatrix}$

- Being $R_{\mathcal{G}}(p) = \dfrac{\partial g_{\mathcal{G}}(p)}{\partial p}$ one obtains

$$R_{\mathcal{G}}(p) = \boxed{\begin{bmatrix} p_1^T - p_2^T & p_2^T - p_1^T & 0 & 0 \\ p_1^T - p_4^T & 0 & 0 & p_4^T - p_1^T \\ 0 & p_2^T - p_3^T & p_3^T - p_2^T & 0 \\ 0 & p_2^T - p_4^T & 0 & p_4^T - p_2^T \\ 0 & 0 & p_3^T - p_4^T & p_4^T - p_3^T \end{bmatrix}}$$

The rigidity eigenvalue

- By embedding in $\mathbb{R}^3$ one obtains $\dim\ker(R_{\mathcal{G}}(p)) = 6$ for a rigid graph
  - The constraint-preserving motions are the 3 translations and 3 rotations around an arbitrary p^* (the motions of a rigid body in 3D space)

- The symmetric rigidity matrix is defined as

$$\mathcal{R} = R_{\mathcal{G}}^T(p) R_{\mathcal{G}}(p) \in \mathbb{R}^{3N \times 3N}$$

- The eigenvalues satisfy $\lambda_1 = \lambda_2 = \cdots = \lambda_6 = 0, \quad \lambda_7 > 0$

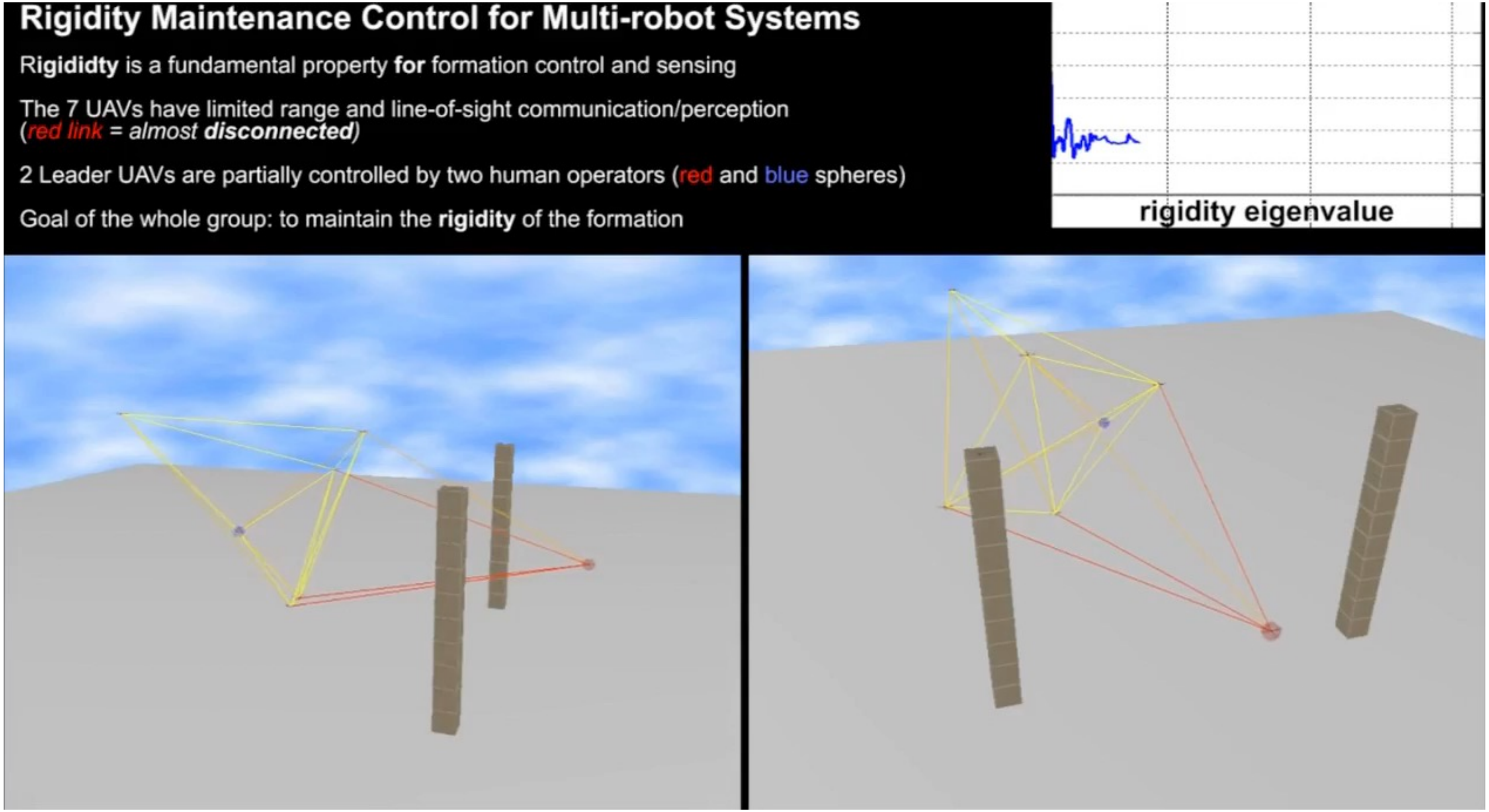
- λ_7 is termed as the rigidity eigenvalue

- The eigenvalue λ_7 can be used as smooth measure of rigidity

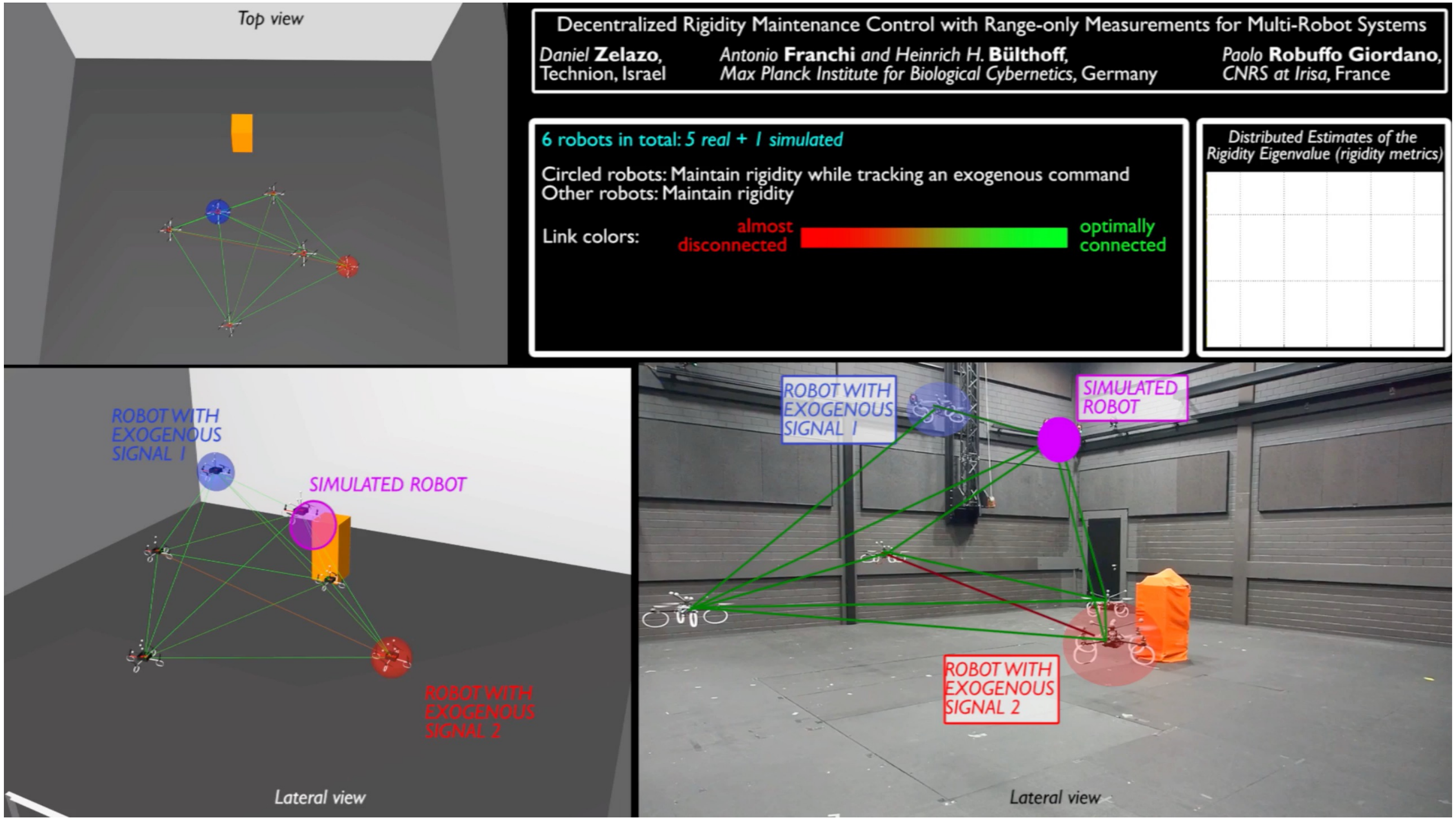
$$\boxed{u_i = -\frac{\partial V^\lambda}{\partial \lambda_7} \frac{\partial \lambda_7}{\partial p_i}}$$

- One can apply the same reasoning as in the case of connectivity to enforce rigidity.

Rigidity Maintenance



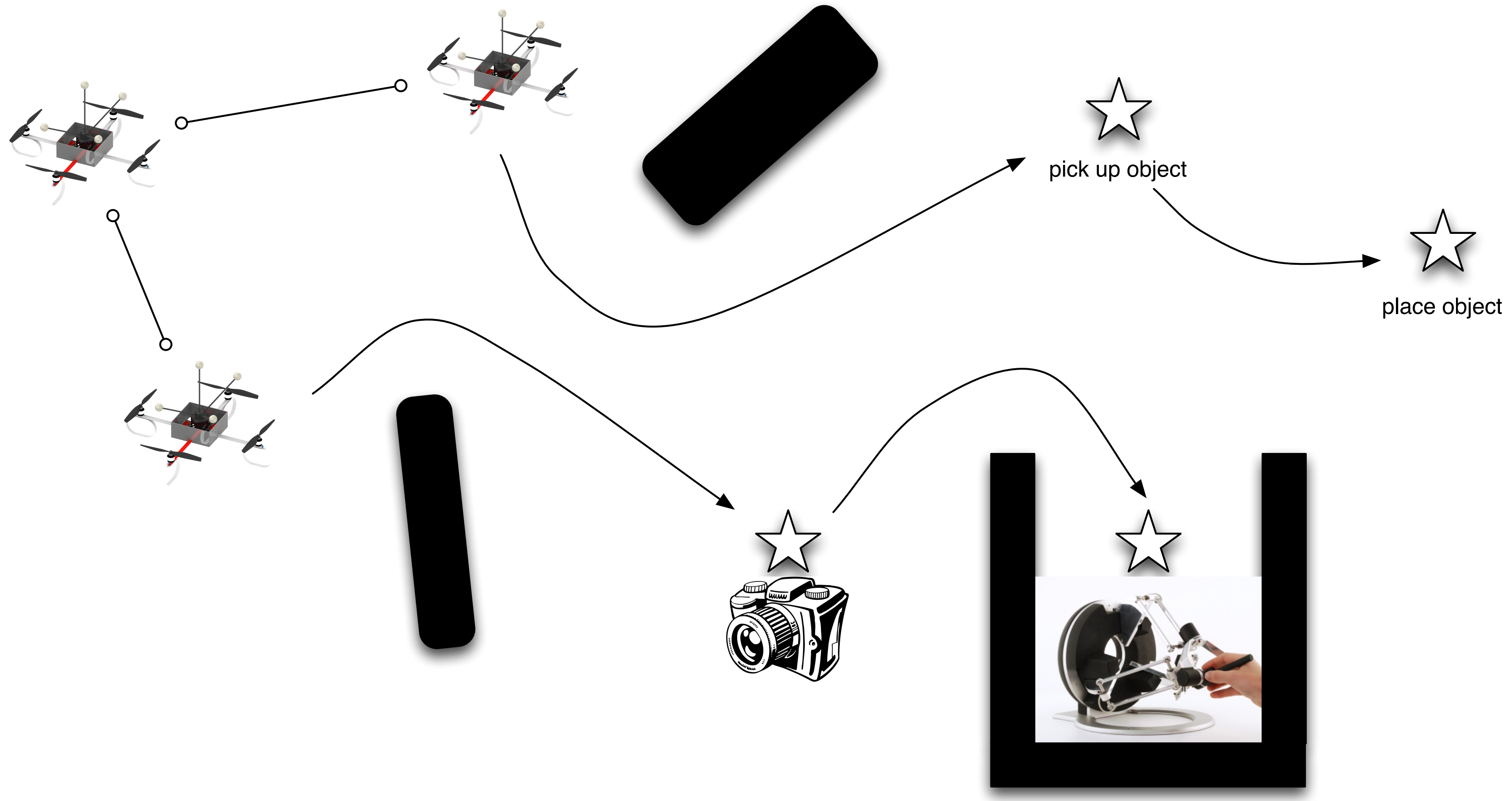
Rigidity Maintenance



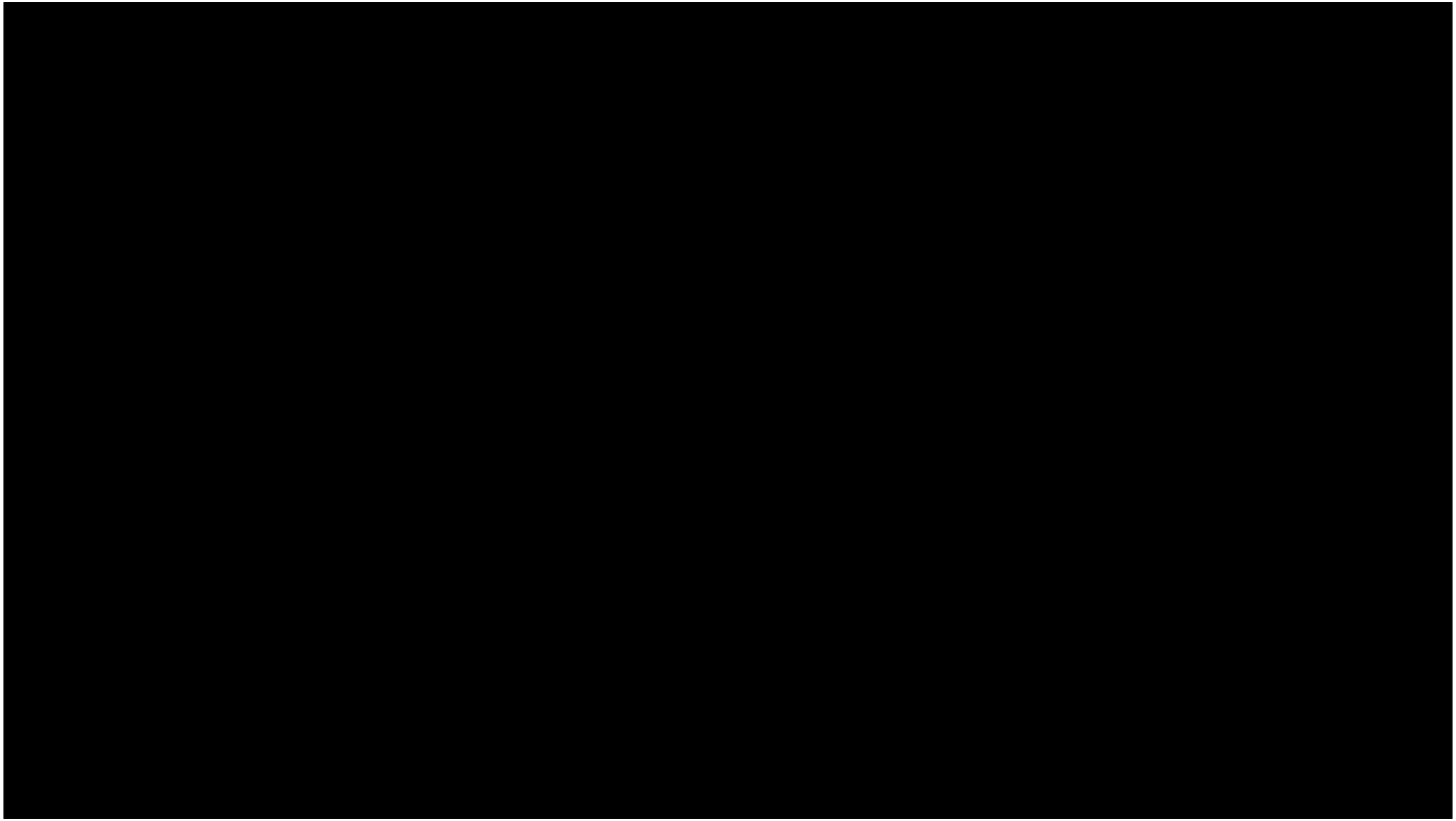
- The quadrotors are maintaining formation rigidity
- This allows them to run a decentralized estimator able to obtain relative positions out of measured relative distances
- Relative positions are then needed by the rigidity controller

Simultaneous Multi-target Exploration with Connectivity Maintenance

- Decentralized Multi-target Exploration and Connectivity Maintenance with a Multi-robot System



Simultaneous Multi-target Exploration with Connectivity Maintenance



Acknowledgments

- The presented material (ideas, theory, applications, experiments) has been mainly conceived and developed by

- Paolo Robuffo Giordano
- Antonio Franchi
- Cristian Secchi
- Daniel Zelazo
- Hyoung Il Son

Main References

- P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, “A Passivity-Based Decentralized Strategy for Generalized Connectivity Maintenance”, International Journal of Robotics Research, 2013

- A. Franchi, C. Secchi, M. Ryll, H. H. Bülthoff, and P. Robuffo Giordano, “Bilateral Shared Control of Multiple Quadrotors” cond. accepted at the IEEE Robotics and Automation Magazine, Special issue on Aerial Robotics and the Quadrotor Platform, 2012

- D. Zelazo, A. Franchi, F. Allgöwer, H. H. Bülthoff, and P. Robuffo Giordano, “Rigidity Maintenance Control for Multi-Robot Systems”, RSS 2012

- A. Franchi, C. Secchi, H. Il Son, H. H. Bülthoff, and P. Robuffo Giordano, “Bilateral Teleoperation of Groups of Mobile Robots with Time-Varying Topology”, IEEE Transactions on Robotics, 2012 (accepted)

- C. Secchi, A. Franchi, H. H. Bülthoff, and P. Robuffo Giordano “Bilateral Teleoperation of a Group of UAVs with Communication Delays and Switching Topology”, ICRA 2012

- P. Robuffo Giordano, A. Franchi, C. Secchi and H. H. Bülthoff, “Experiments of Passivity-Based Bilateral Aerial Teleoperation of a Group of UAVs with Decentralized Velocity Synchronization”, IROS 2011

- P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, “Bilateral Teleoperation of Groups of UAVs with Decentralized Connectivity Maintenance”, RSS 2011

- A. Franchi, P. Robuffo Giordano, C. Secchi, H. I. Son, and H. H. Bülthoff, “A Passivity-Based Decentralized Approach for the Bilateral Teleoperation of a Group of UAVs with Switching Topology”, ICRA 2011