Deformation-based Shape Control with a Multirobot System

DIPARTIMENTO DI INGEGNERIA INFORMATICA
AUTOMATICA E GESTIONALE ANTONIO RUBERTI

SAPIENZA
UNIVERSITÀ DI ROMA

Di Tucci Roberta
Marroni Tommaso

Introduction

Goal: design a method to control the shape of the robotic team.

Possible application: cooperative manipulation of an object in 2D space (minimizing the deformation w.r.t. a desired object state).

- Robot motions are based on the descent along the gradient of a metric that expresses the difference between current configuration and desired shape.

General problem statement

- Consider a set of N robots, with kinematics:

$$\dot{\mathbf{q}}_\mathbf{i} = \mathbf{u}_\mathbf{i}, \quad \forall i \in \mathcal{N}$$

- *Desired geometry*: constant over time, defined by a set of points $\boldsymbol{c}_i \in \mathbb{R}^2$.

- *Shape control*: make $\{\boldsymbol{q}_i\}$ satisfy:

$$\mathbf{q}_\mathbf{i} = s\mathbf{R}\mathbf{c}_\mathbf{i} + \mathbf{t} \; \forall i \in \mathcal{N}$$

Team has the same shape as the desired geometry.

Control strategy

Procrustes shape-fitting problem

$$\gamma_g = \frac{1}{2} \sum_{i \in \mathcal{N}} ||\mathbf{q_i} - \mathbf{H_g c_i} + \mathbf{t_g}||^2$$

- γ_g is a measure of the SSD (Sum of Squared Differences) error between the $\{\boldsymbol{q}_i\}$ and the $\{\boldsymbol{c}_i\}$ with $\{\boldsymbol{c}_i\}$ rotated and scaled by $\boldsymbol{H}_g$, and translated by $\boldsymbol{t}_g$.
- *Procrustes shape-fitting problem*: finding the optimal $\boldsymbol{H}_g$ and $\boldsymbol{t}_g$ minimizing $\boldsymbol{\gamma}_g$.

  - $\boldsymbol{t}_g$ = 0
  - Replace $\boldsymbol{q}_i$ with $\boldsymbol{q}_{io}$ and $\boldsymbol{c}_i$ with $\boldsymbol{c}_{io}$.
  - $\boldsymbol{H}_g$ = $\boldsymbol{H}$

$$\mathbf{H} = \begin{bmatrix} h_1 & -h_2 \\ h_2 & h_1 \end{bmatrix}$$

$$h_1 = \frac{1}{c_s} \sum_{i \in \mathcal{N}} \mathbf{q_{io}^T c_{io}}$$

$$h_2 = \frac{1}{c_s} \sum_{i \in \mathcal{N}} \mathbf{q_{io}^T c_{io}^{\perp}}$$

Control strategy

γ cost function

- Obtained cost function:

$$\gamma = \frac{1}{2} \sum_{i \in \mathcal{N}} ||\mathbf{q_{io}} - \mathbf{H c_{io}}||^2$$

- γ interpreted as measure of deformation relative to desired shape.
- Note that $\{\boldsymbol{q}_i\}$ is equal to $\{\boldsymbol{c}_i\}$ up to translation, rotation and scaling if and only if $\gamma = 0$.
- $\boldsymbol{H}$ can be defined in terms of scaling s_h and rotation $\boldsymbol{R}_h$: $\boldsymbol{H} = s_h \boldsymbol{R}_h$.

$$\mathbf{R_h} = \begin{bmatrix} \cos(\alpha_h) & -\sin(\alpha_h) \\ \sin(\alpha_h) & \cos(\alpha_h) \end{bmatrix} \qquad s_h = \sqrt{{h_1}^2 + {h_2}^2}$$

$$\alpha_h = atan2(h_2, h_1)$$

Control strategy

Control law

- **Idea**: use the opposite of the gradient of γ in the controller in order to minimize deformation w.r.t. desired shape.

$$\nabla_{\mathbf{q_i}}\gamma = \frac{\partial \gamma}{\partial \mathbf{q_i}} = \sum_{j\in\mathcal{N}} (\mathbf{q_{jo}} - \mathbf{Hc_{jo}})^T \frac{\partial \mathbf{q_{jo}}}{\partial \mathbf{q_i}} =$$

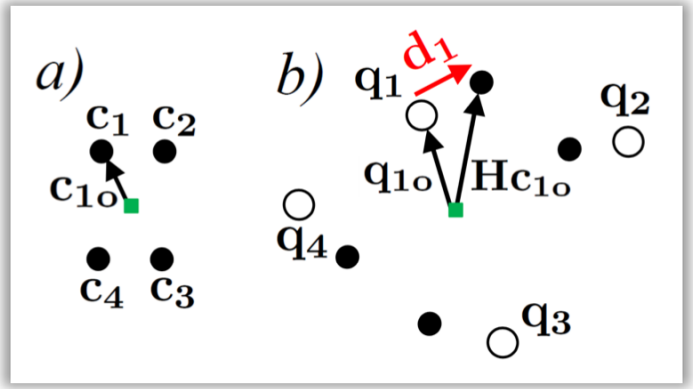
$$(1-\frac{1}{N})(\mathbf{q_{io}} - \mathbf{Hc_{io}}) - \frac{1}{N}\sum_{\substack{j\in\mathcal{N} \\ j\neq i}} \mathbf{q_{jo}} - \mathbf{Hc_{jo}} = \mathbf{q_{io}} - \mathbf{Hc_{io}}$$

- The control law of each robot is defined by:

$$\mathbf{u_i} = k_i \mathbf{d_i}$$

with:

$$\mathbf{d_i} = -\nabla_{\mathbf{q_i}}\gamma = \mathbf{q_{oi}} - \mathbf{Hc_{oi}}$$



Controller analysis and properties

Properties and remarks

- **Theorem**: under the action of the previous control law the robots converge exponentially to a configuration in which they form the desired shape and stay static.
  - …can be proved using Lyapunov and LaSalle.
- Note that γ can be expressed as:

$$\gamma = \frac{N}{2}({s_q}^2 - {s_d}^2)$$

where

$$s_d = \sqrt{\frac{1}{N}\sum_{i\in\mathcal{N}} ||\mathbf{Hc_{io}}||^2} = s_h\sqrt{\frac{c_s}{N}}$$ (Destination configuration size)

$$s_q = \sqrt{\frac{1}{N}\sum_{i\in\mathcal{N}} ||\mathbf{q_{io}}||^2}$$ (Current configuration size)

Controller analysis and properties

Properties and remarks

- γ expresses the difference in shape but it also encapsulates the difference in size.

- It holds that $s_d \leq s_q$ for any current and desired configuration.
  - Controller moves the robots towards a configuration s.t. its size is never larger then the current one.
    - Energy of the system is mantained and divergent behaviors are avoided (no object overstretching)

- If all robots use same gain k_c, then $\boldsymbol{H}$ remains constant (s_h and $\boldsymbol{R}_h$ constant).

Variants of the controller

Team motions in the rotation and scaling nullspaces.

- **Rotation-preserving motions**: motions such that $\dot{\boldsymbol{R}}_h = 0$.
- **Proposition**: a motion strategy where the robots implements the control law that follows

$$\mathbf{u_i} = k_c(\mathbf{q_{oi}} - w_i(t)\mathbf{H}\mathbf{c_{oi}}),\ w_i(t) \in \mathbb{R}_{>0},\ \forall i \in \mathcal{N}$$

is rotation-preserving.

- **Scale-preserving motions**: motions such that $\dot{s}_h = 0$.
- **Proposition**: a motion strategy where the robots implements the control law that follows

$$\mathbf{u_i} = k_c((\mathbf{q_{oi}} - \mathbf{H}\mathbf{c_{oi}}) + z_i(t)\mathbf{H}\mathbf{c}_{\mathbf{oi}}^{\perp}),\ z_i(t) \in \mathbb{R},\ \forall i \in \mathcal{N}$$

is scale-preserving.

Variants of the controller

Fixing team's size and orientation using a single robot

- If all robots use the same control gain k_c, robots $i = 2, \dots, N$ implement the standard control law, and robot $i = 1$ uses

$$\mathbf{u_1} = k_c \mathbf{d_1}, \text{ with } \mathbf{d_1} = \mathbf{q_{o1}} - \mathbf{H_1 c_{o1}}$$

  then $\boldsymbol{H}$ tracks $\boldsymbol{H}_1$ according to $\dot{\boldsymbol{H}} = -k_{f1}(\boldsymbol{H} - \boldsymbol{H}_1)$.
  - With $k_{f1} = k_c \|\boldsymbol{c_{1o}}\|^2 / c_s$

- Thus:
  - If $\boldsymbol{H}_1$ is constant, $\boldsymbol{H}$ converges to it exponentially;
  - If $\boldsymbol{H}_1$ is time-varying, $\boldsymbol{H}$ tracks its variation;
  - The special robot can in this way steer the value of $\boldsymbol{H}$.

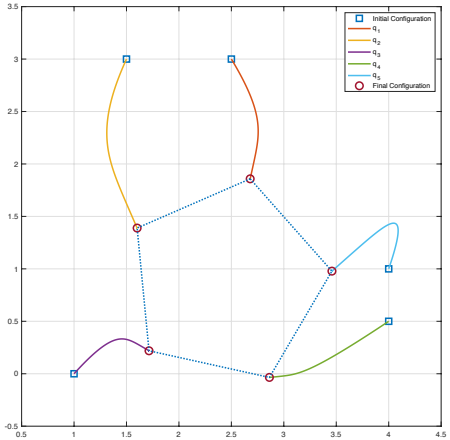
Variants of the controller

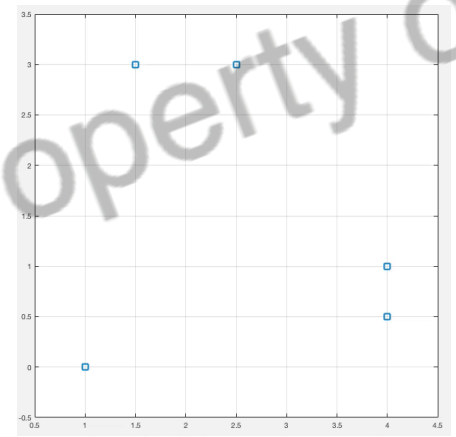
Fixing team's size and orientation using a single robot: strategies S1 and S2

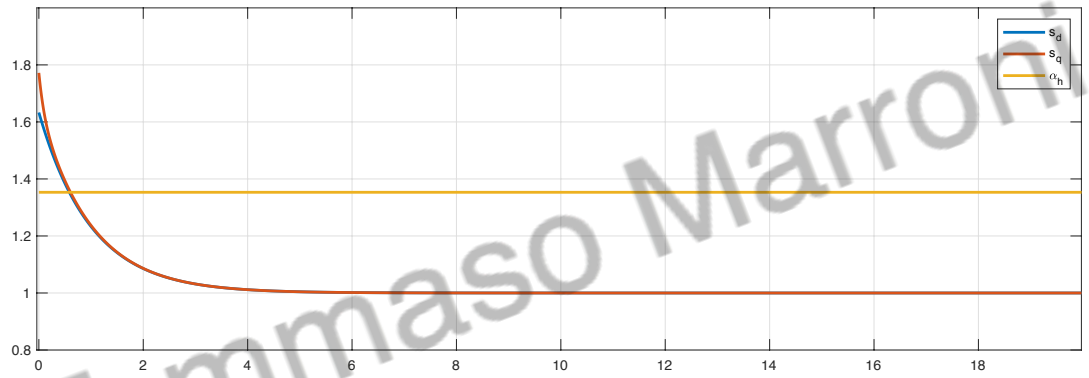
- Two possible strategies based on the previous considerations:
- *S1*: Drive team size s_h towards a desired value $s_h = s_1$ without changing R_h.
  - Set: $w_1(t) = \frac{s_1}{s_h}$, $s_h > 0$; and $w_i(t) = 1$ for $i = 2, \dots, N$.
- *S2*: Rotate team towards a desired orientation $R_h = R(\alpha_1)$ without changing s_h.
  - Set $z_1(t) = k_{r1}(\alpha_h - \alpha_1)$, $k_{r1} > 0$ and $z_i(t) = 0$ for $i = 2, \dots, N$.
- Note that S1 and S2 comes from the previous proposition.

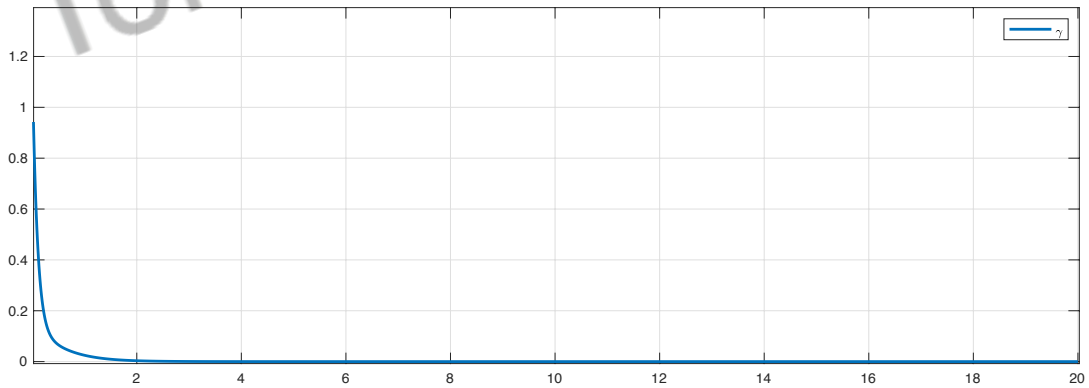
Simulations

Simulation 1 (S1)





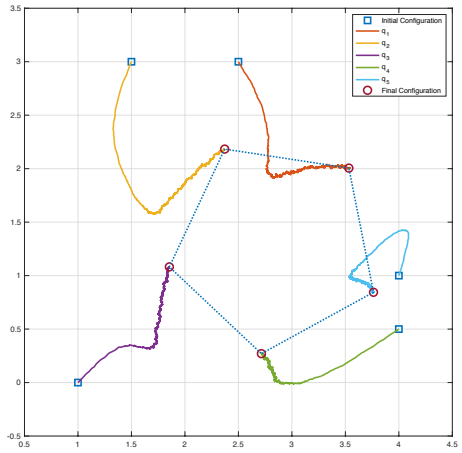


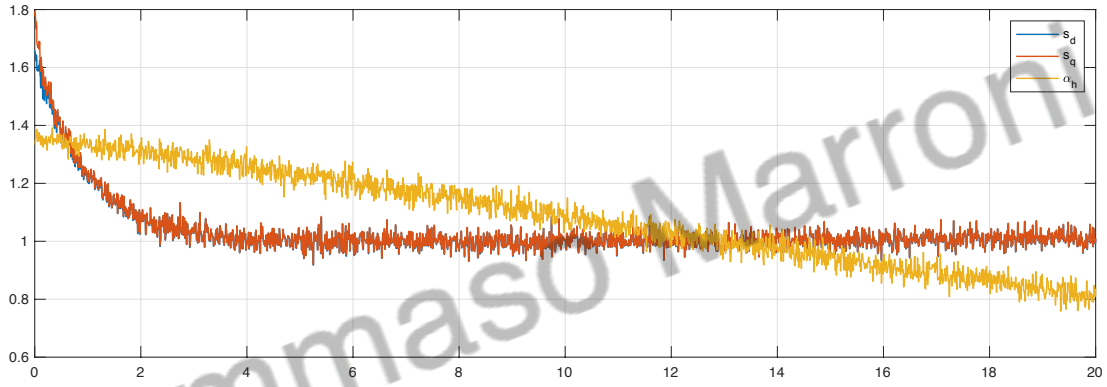


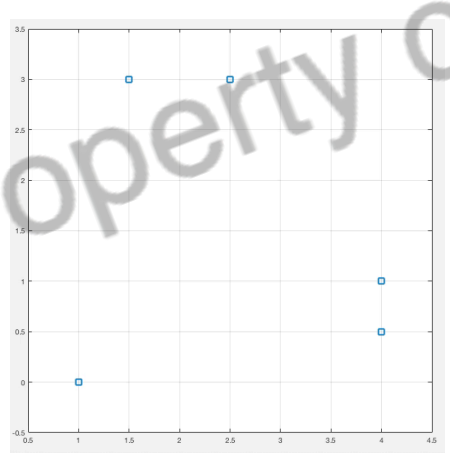
$k_c = 5,\ s_1 = 1, T = 20$

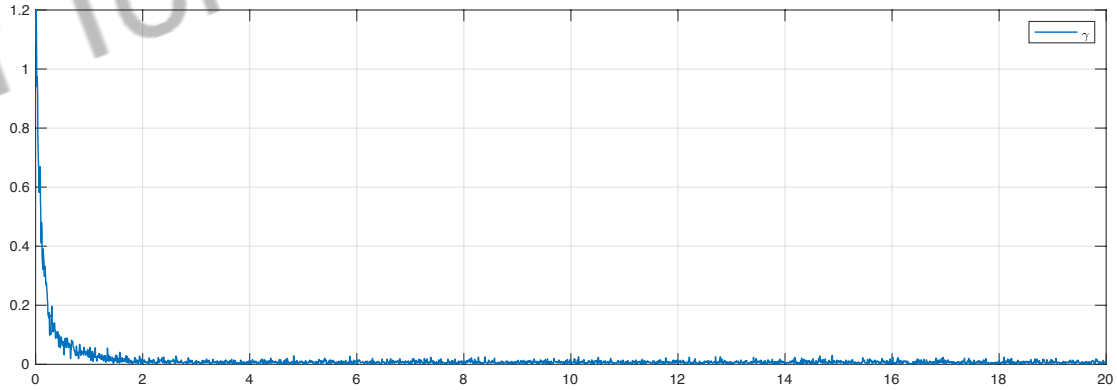
Simulations

Simulation 2 *(S1 + noise + error)*





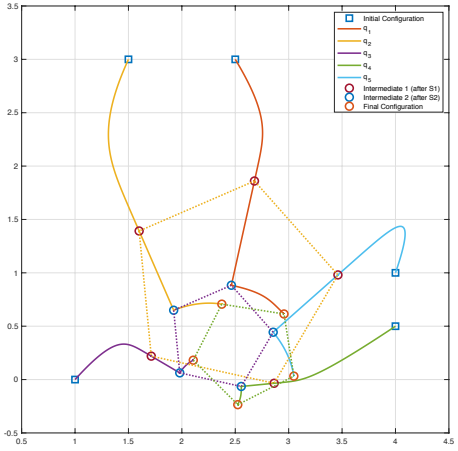


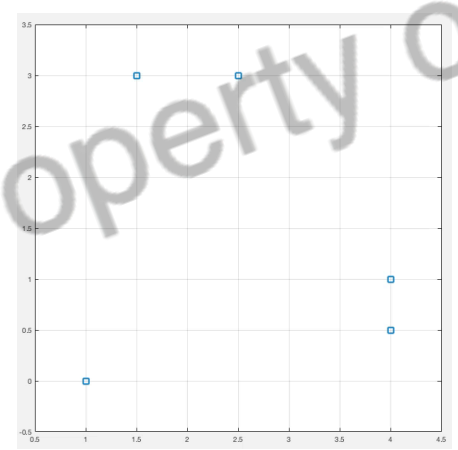


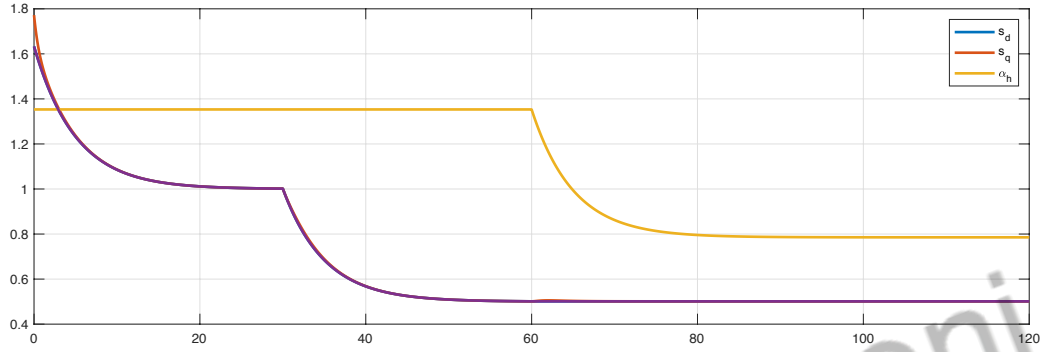
$k_c = 5,\ s_1 = 1, T = 20$

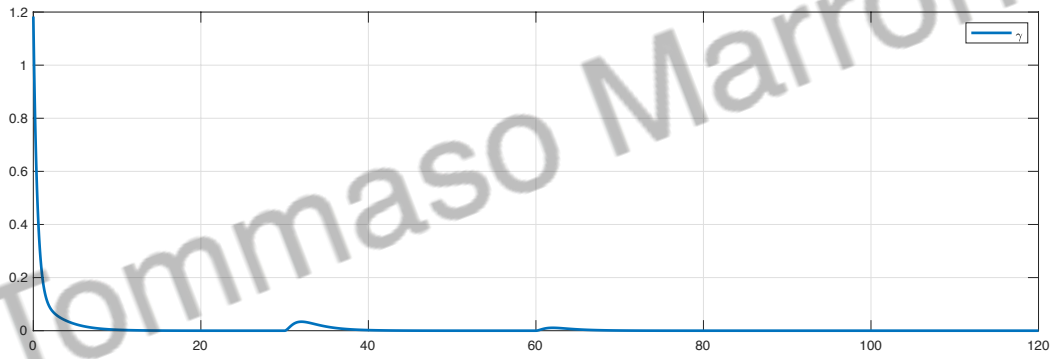
Simulations

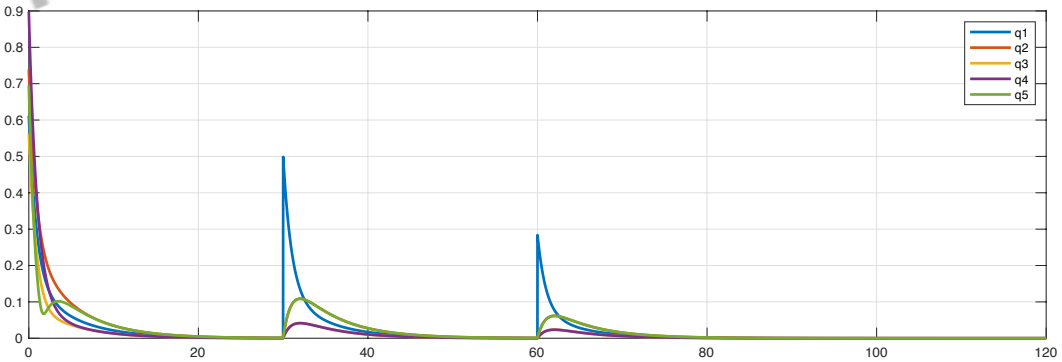
Simulation 3 (S1,S1,S2)











$$k_c = 1, k_r = 1, s_1^{t_0} = 1, s_1^{t_1} = 0.5, \alpha_1 = \frac{\pi}{4}, t_1 = 30, t_2 = 60, T = 20$$

Variants of the controller

Orientation control

- Orientation control law:

$$\dot{\theta}_i = k_{ai}(\theta_i^d - \theta_i)$$

where θ_i^d is the is the desired orientation and $k_{ai} > 0$.

- Simulation 4: $\theta_i^d = 0$, $i = 1, \dots, N$;

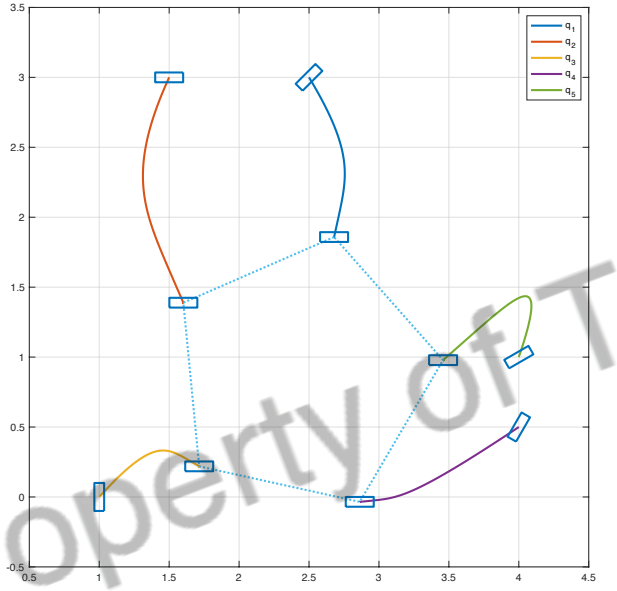
- Simulation 5: $\theta_1^d = \angle(\overline{\boldsymbol{q_{2,1}}})$; $\theta_N^d = \angle(\overline{\boldsymbol{q_{N,N-1}}})$; $\theta_i^d = \angle(\overline{\boldsymbol{q_{i,i-1}}} + \overline{\boldsymbol{q_{i+1,i}}})$ for $i = 2, \dots, N-1$.

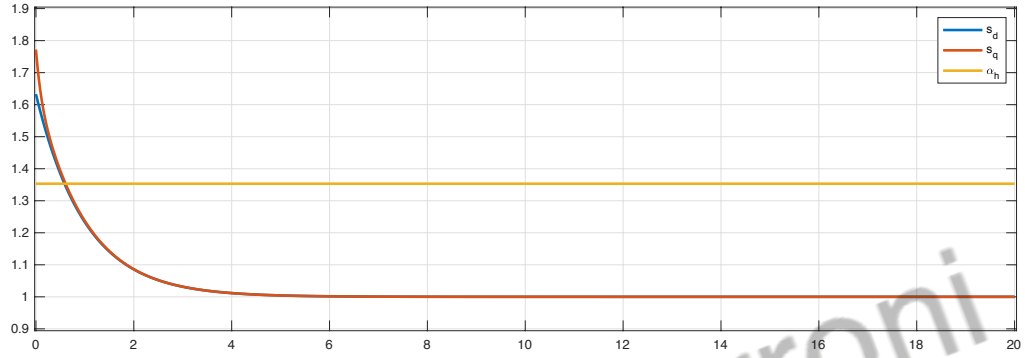
  - Simple strategy to avoid sharp curvature of an hypothetical elastic rod grasped by the robots.

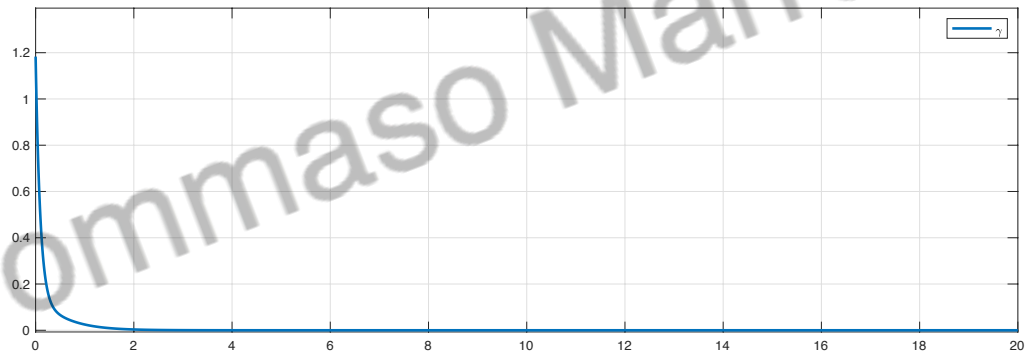
Simulations

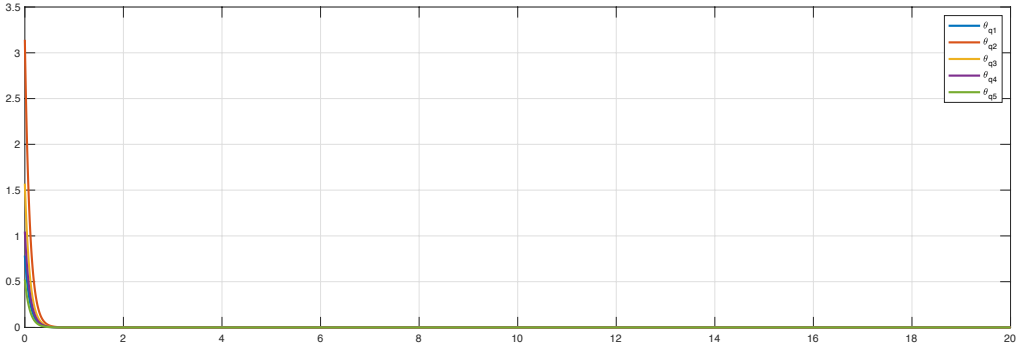
Simulation 4

(S1 + orientation)







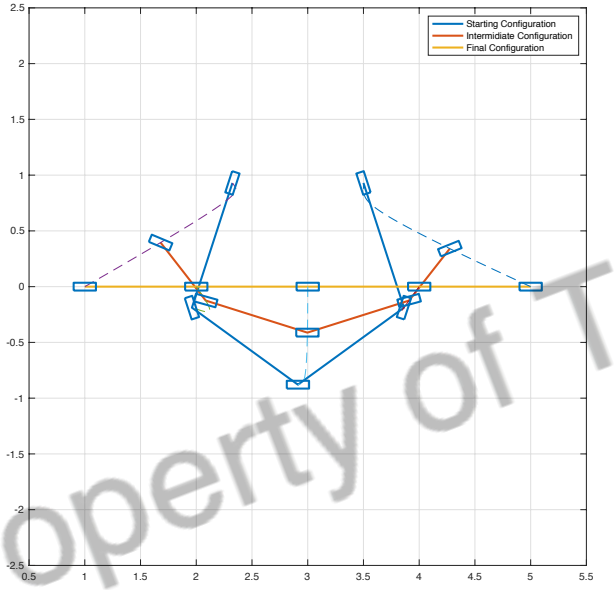


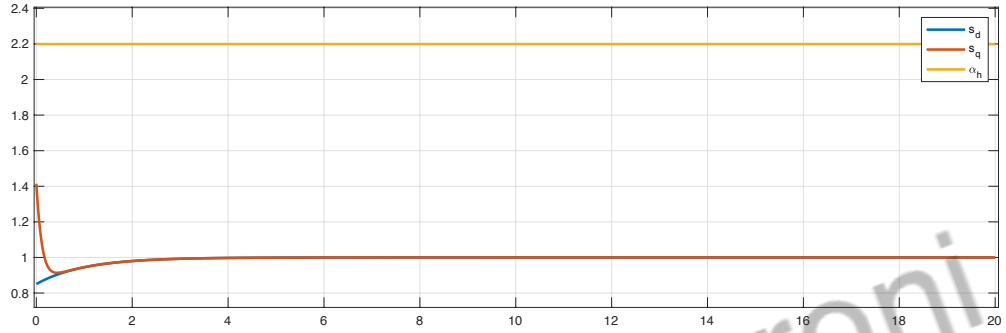
$k_c = 5, k_\theta = 10, s_1 = 1, T = 20$

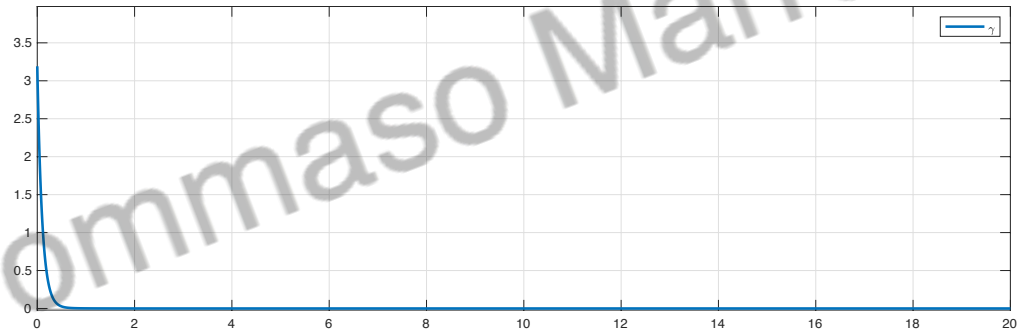
Simulations

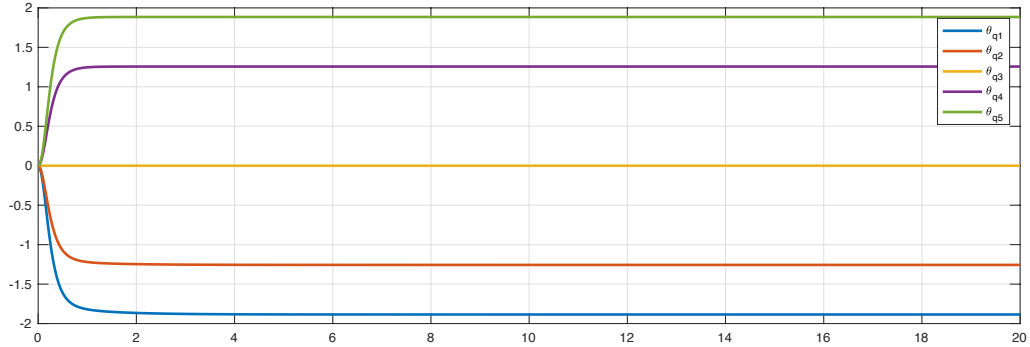
Simulation 5

(S1 + orientation)









$k_c = 5, k_\theta = 10, s_1 = 1, T = 20$

Conclusion

- Using γ (differentiable and simple computationally) the team of robots can be controlled with a simple control law.
  - Fundamental parameters (like size or rotation) can be steered and controlled.

- Limitations and issues not considered:
  - Controller needs knowledge about all the robots;
  - Could be necessary to use additonal planning if desired shape is very different from initial one;