Tampere University

Homework 10

M.W.S. Atman, A. Gusrialdi

April 3, 2023
Deadline: 18.04.2023 (11:59 hours)

Consider 4 omnidirectional robots with each robot dynamic is given by

$$\dot{x}_i = u_i, \quad i \in \{1, \ldots, 4\} \tag{1}$$

where $x_i = [p_{x,i}\ p_{y,i}]^T \in \mathbb{R}^2$ represents the position of the robot and $u_i = [u_{x,i}\ u_{y,i}]^T \in \mathbb{R}^2$ denotes the control input to be designed.

Each robot i is initiated at location $x_i(0)$ and need to move toward its own goal position $x_{g,i}$ by following a go-to-goal controller $u_{gtg,i} = (x_{g,i} - x_i)$.
Let us denote the combined state, control input, and goal positions of all robots as follow

$$x = [x_1^T \cdots x_4^T]^T \in \mathbb{R}^8, \quad u = [u_1^T \cdots u_4^T]^T \in \mathbb{R}^8, \quad x_g = [x_{g,1}^T \cdots x_{g,4}^T]^T \in \mathbb{R}^8$$

with robots initial and goal position are at

$$x(0) = [-0.6\ 0.6\ \ -0.2\ 0.4\ \ 0.2\ 0.5\ \ 0.6\ 0.6]^T m,$$

$$x_g = [1.2\ 2.5\ \ -0.4\ 2.5\ \ 0.4\ 2.5\ \ -1.2\ 2.5]^T m.$$

Note: Unless specified, for all simulation use time sampling of 10ms. In each of your answer provide the simulation results of the XY-trajectory plot of the robot, the **time series of the distance between all pairs of robots** ($||x_i - x_j||$), and the comparison of the nominal and optimal control input, i.e. $u_{gtg,i}$ vs u_i.

1 Problem 1 (25 Points)

Consider a **centralized** optimization-based safety control as follow

$$\begin{aligned} u = &\underset{u_i,\ i=\{1,\ldots,4\}}{\arg\min} \sum_{i=1}^{4} \|u_{gtg,i} - u_i\|^2 \\ &\text{s.t.} \ \left(\frac{\partial h_o(x_i,x_j)}{\partial x_i}\right)^T u_i + \left(\frac{\partial h_o^3(x_i,x_j)}{\partial x_j}\right)^T u_j \geq -2\gamma h_o^3(x_i,x_j), \quad \forall j \neq i \end{aligned} . \tag{2}$$

with $\gamma = 10$ and $h_o(x_i, x_j) = ||x_i - x_j||^2 - 0.2^2$.

1. Given the general form of quadratic programming:

$$\begin{aligned} u = \arg\min_u \ &\frac{1}{2} u^T Q u + c^T u \\ \text{s.t.} \ &Hu \leq b \end{aligned} \tag{3}$$

   compute Q, c, H, and b. (10 points)

2. Design a simulation in Matlab and implement the controller (2) (and (3)) to safely control all robots towards their goals without colliding with each other. (15 points)

2 Problem 2 (35 Points)

With a similar setup to Problem 1, consider a **distributed** optimization-based safety control as follow

$$\begin{aligned} u_i = &\arg\min_{u_i} \|u_{gtg,i} - u_i\|^2 \\ &\text{s.t.} \ \left(\frac{\partial h_o(x_i,x_j)}{\partial x_i}\right)^T u_i \geq -\gamma h_o^3(x_i,x_j), \quad \forall j \in \mathcal{N}_i^s \end{aligned} . \tag{4}$$

with $\mathcal{N}_i^s = \{j,\ j \neq i \mid ||x_i - x_j||^2 \leq 1^2\}$

1. Given the general form of quadratic programming for distributed case:

$$\begin{aligned} u_i = \arg\min_{u_i} \ &\frac{1}{2} u_i^T Q_i u_i + c_i^T u_i \\ \text{s.t.} \ &H_i u_i \leq b_i. \end{aligned} \tag{5}$$

   Assume a case where robot 1 is nearby all other robot, i.e., $\mathcal{N}_1^s = \{2, 3, 4\}$, compute Q_1, c_1, H_1, and b_1. (10 points)

2. Design a simulation in Matlab and implement the controller (4) (and (5)) to safely control all robots towards their goals without colliding with each other. (15 points)

3. Compare the simulation results in Problem 1 and 2. Provide a discussion on advantage and disadvantages of each approach (centralized vs. distributed). (10 points)

3 Problem 3 (40 Points)

In this problem we are considering additional constraints where all agents need to maintain the network connectivity when moving. Hence, the controller in (4) is updated into

$$\begin{aligned} u_i = \arg \min_{u_i} \; & \|u_{gtg,i} - u_i\|^2 \\ \text{s.t.} \; & \left(\frac{\partial h_o(x_i, x_j)}{\partial x_i}\right)^T u_i \geq -\gamma h_o^3(x_i, x_j), \quad \forall j \in \mathcal{N}_i^s. \\ & \left(\frac{\partial h_c(x_i, x_j)}{\partial x_i}\right)^T u_i \geq -\gamma h_c^3(x_i, x_j), \quad \forall j \in \mathcal{N}_i^c \end{aligned} \tag{6}$$

with $h_c(x_i, x_j) = 0.9^2 - ||x_i - x_j||^2$ and $\mathcal{N}_i^c$ denotes the robot i's neighbors defined by the prescribed graph communication topology.

1. Given the general form of quadratic programming for distributed case in (5). Assume a case where robot 1 is nearby all other robot, i.e., $\mathcal{N}_1^s = \{2, 3, 4\}$, and its communication neighbors are specified by $\mathcal{N}_1^c = \{2, 3, 4\}$. Compute the new Q_1, c_1, H_1, and b_1. (10 points)

2. Design a simulation in Matlab and implement the controller (6) (and (5)) to safely control all robots towards their goals without colliding with each other while ensuring a complete graph communication topology, i.e., each robot connected to all other robots. When implemented correctly, this scenario will hinder several robots to reach their goals. Try to explain why it happens. (15 points)

3. Propose an adjacency matrix that will allow each robot to reach their goals while maintaining network connectivity *(Hint: spanning tree)*. Verify the results via Matlab simulation. (15 points)