FIELD AND SERVICE ROBOTICS (FSR) – a.y. 2023/2024
University of Naples Federico II
Department of Electrical Engineering and Information Technology

Instructor: Fabio Ruggiero, Ph.D. [fabio.ruggiero@unina.it]
Assistant: Riccardo Aliotta, M.Sc. [riccardo.aliotta@unina.it]
Assistant: Francesco Cufino, M.Sc. [francesco.cufino@unina.it]

[Updated 16/04/2024]

HOMEWORK n. 2

1. State whether each of following distributions are involutive or not, and briefly justify your answer. If possible, find the annihilator for each distribution.

   a. $\Delta_1 = \left\{ \begin{bmatrix} 3x_1 \\ 0 \\ -1 \end{bmatrix} \right\}, \boldsymbol{U} \in \mathbb{R}^3$

   b. $\Delta_2 = \left\{ \begin{bmatrix} 1 \\ 0 \\ x_2 \end{bmatrix}, \begin{bmatrix} 0 \\ -\alpha \\ x_1 \end{bmatrix} \right\}, \boldsymbol{U} \in \mathbb{R}^3$ with α the last digit of your matriculation number.

   c. $\Delta_3 = \left\{ \begin{bmatrix} 2x_3 \\ -1 \\ 0 \end{bmatrix}, \begin{bmatrix} x_2 \\ x_1 \\ 1 \end{bmatrix} \right\}, \boldsymbol{U} \in \mathbb{R}^3$

2. Consider an omnidirectional mobile robot having 3 Mecanum wheels placed at the vertices of an equilateral triangle, each oriented in the direction orthogonal to the bisectrix of its angle. Let x and y be the Cartesian coordinates of the center of the robot, θ the vehicle orientation and α, β, γ represent the angle of rotation of each wheel around its axis. Also, denote by r the radius of the wheels and by l the distance between the centre of the robot and the centre of each wheel. This mechanical system is subject to the following Pfaffian constraints, where $q = [x \quad y \quad \theta \quad \alpha \quad \beta \quad \gamma]^T$

$$A^T(q)\dot{q} = \begin{bmatrix} \frac{\sqrt{3}}{2}\cos\theta - \frac{1}{2}\sin\theta & \frac{1}{2}\cos\theta + \frac{\sqrt{3}}{2}\sin\theta & l & r & 0 & 0 \\ \sin\theta & -\cos\theta & l & 0 & r & 0 \\ -\frac{\sqrt{3}}{2}\cos\theta - \frac{1}{2}\sin\theta & \frac{1}{2}\cos\theta - \frac{\sqrt{3}}{2}\sin\theta & l & 0 & 0 & r \end{bmatrix} \dot{q} = 0_6.$$

Compute a kinematic model of such an omnidirectional robot and show whether this system is holonomic or not. *[Hint: Use Matlab symbolic toolbox and the null command to ease your work. Do not care about the physical meaning of the kinematic inputs.]*

3. Implement via software the path planning algorithm for a unicycle based on a cubic Cartesian polynomial. Plan a path leading the robot from the configuration $q_i = [x_i \quad y_i \quad \theta_i]^T = [0 \quad 0 \quad 0]^T$, to a random configuration $q_f = [x_f \quad y_f \quad \theta_f]^T$ generated automatically by the code such that $\|q_f - q_i\| = 1$. Then, determine a timing law over the path to satisfy the following velocity bounds $|v(t)| \leq 2$ m/s and $|\omega(t)| \leq 1$ rad/s. [*Hint: For the final configuration, use the command rand(1,3): save the result in a vector and divide it by its norm.*]

4. Given the trajectory in the previous point, implement via software an input/output linearization control approach to control the unicycle's position. Adjust the trajectory accordingly to fit the desired coordinates of the reference point B along the sagittal axis, whose distance to the wheel's center it is up to you.

5. Implement via software the unicycle posture regulator based on polar coordinates, with the state feedback computed through the Runge-Kutta odometric localization method. Starting and final configurations are $q_i = [x_i \quad y_i \quad \theta_i]^T = \left[\alpha + 1 \quad 1 \quad {}^{\pi}/_{4}\right]^T$, with α the last digit of your matriculation number, and $q_f = [x_f \quad y_f \quad \theta_f]^T = [0 \quad 0 \quad 0]^T$.

NOTE: It is worth recalling not to report theory in the report. **Put all the plots you think are the most important to understand the performance of the code you implemented, and critically comment on the**

results. Attach the code with your submission in a ZIP file. If you overcome the submission limit on Moodle, you may link in the report a GitHub, Dropbox, or Google Drive link (make these links public, if possible, to avoid waiting for permission to download the files).
The software is left free. Matlab is anyway suggested to save time. Notice that the Simulink Robotics Toolbox already have the block with the unicycle kinematic model implemented (Robotics Toolbox -> Mobile Robot Algorithms).