Control barrier function-enabled human-in-the-loop control for multi-robot systems

Victor Nan Fernandez-Ayala
vnfa@kth.se

January 13, 2022

Table of Contents

Introduction

This part was used for the MSc Thesis registration process and each section was one of the questions to fill in.

1.1 Background

Autonomous multi-robot systems have found many real-world applications in factory settings, rescue tasks, and light shows. Albeit these successful applications, the multi-robot system is usually pre-programmed with limited flexibility for online adaptation. Having a human in the loop would provide additional flexibility such as handling unexpected situations, detecting and correcting bad behaviors, and supporting the automated decision making. In this thesis work, we study how to incorporate human commands with multi-robot systems with an intention to allow the systems to execute human commands whenever safe, and implement the designed scheme on real robots. Control barrier functions (CBFs), as a convenient modular-design tool, will be mainly explored.

This work will be developed as a continuation of the paper titled "Distributed implementation of control barrier functions for multi-agent systems" developed by Xiao Tan and Dimos V. Dimarogonas at the Division of Decision and Control Systems at KTH. In addition, it also has synergies with the COIN and CANOPIES projects also developed at KTH (and other institutions), where multi-robot systems with HuIL (Human in the Loop) are also explored.

1.2 Research question

The main objective of the project is to design a CBF-enabled mixer module to combine existing multi-robot coordination algorithms with human commands while making sure that the safety constraints are never violated, and validate the theoretical results.

1.3 Hypothesis

In this thesis work, we are expected to identify safety constraints for most of existing multi-robot coordination protocols, and enforce these safety constraints in a control barrier function-enabled mixer module, to be designed in this thesis work, that takes in the commands from both the planner and the human operator. The mixer module is expected to be designed that prioritize the commands from human operator as long as the safety constraint is not violated. Otherwise, the mixer module filters the commands and send out a safe command to multi-robot systems. The underlying multi-robot tasks (trajectory tracking, formation control, coverage control, etc) are expected to be achieved whenever feasible.

1.4 Research method

The process will be the following:

1. Get familiar with several classic multi-robot coordination algorithms, their respective assumptions and control barrier functions;
2. Perform a literature review on human-in-the-loop control for multi-robot systems;
3. Design the CBF-enabled mixer and analyze the closed-loop system performance through classical control methods and mathematical proofs;
4. Implement the multi-robot coordination algorithms and the designed CBF-enabled mixer module on multiple mobile robots, and test them both with simulations using ROS (robotic operating system) and experiments at the SML (Smart Mobility Lab) at KTH.

Formation control

In this first chapter, a multi-robot formation keeping algorithm will be created with as many robots as wanted (currently max. of 5 to ensure convergence due to hardware constraints) to form any formation with any neighbour configuration while keeping a minimum distance between them to secure Communication Maintenance (CM) and Obstacle Avoidance (OA). For the CM, a minimum distance between each robot and its neighbours will always be kept so that they are close enough to keep the communication open; while for OA, the same will be done by making sure there is enough distance between each robot and its neighbours to prevent possible collisions.

Several modifications will be tried next, like adding a HuIL control to one of the agents, introducing a CBF to ensure the robots are always inside the SML arena or inside a custom wedge shape or adding adding a new robot/human which the whole formation needs to avoid.

Finally, it is important to mention that this is a 2D problem since the mobile robot platform NEXUS will be used to simulate and test the algorithms. This allows us, in combination with the Qualisys Motion Capture system to have accurate positional data and control directly the x and y velocity of each robot thanks to its mecanum wheels. A more complex algorithm could be developed using common wheels and the forward speed and the wheel axis rotation as control parameters, but the results will be the same with a more complex system dynamics.

2.1 Optimal centralized version

Let's assume a K-shaped configuration with 5 agents ordered in the following way:

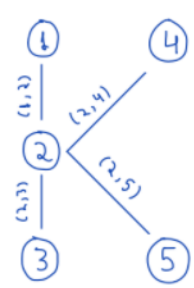


Figure 2.1: K-shaped formation with 5 robots

Every agent has a position determined by the row vector $p_i = [x_i, y_i]$ and a controller $u_i = [u_{ix}, u_{iy}]$, where $i = 1, 2, ..., N$ and N is the number of agents, in this case being 5. Every edge is represented by the tuple (i, j) and since we are computing the centralized version the controllers will only be calculated once. Note that $(i, j) = (j, i)$ and that the following convention is used: $i \leq j$.

The general formula for the controller is the following:

$$\begin{aligned} \min_{u} \quad & ||u - u_{nom}||^2 \\ \text{s.t.} \quad & \Delta h^T f + \Delta h^T g u \geq -\alpha(h) \end{aligned} \tag{2.1}$$

where the nominal controller is the formation controller and the constraints represent the safety parameters. Particularly, h is the CBF function that defines the safety region, such that as long as $h \geq 0$, this will be satisfied. The system dynamics are:

$$\text{for each agent } i = 1, 2, ..., N \Rightarrow \dot{p}_i = f + g u_i = u_i$$

$$\text{so } \dot{p} = \begin{bmatrix} \dot{p}_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} \Rightarrow \text{ for two dimensions x and y } \Rightarrow \begin{bmatrix} \dot{x}_1 & \dot{y}_1 \\ \dot{x}_2 & \dot{y}_2 \\ \dot{x}_3 & \dot{y}_3 \\ \dot{x}_4 & \dot{y}_4 \\ \dot{x}_5 & \dot{y}_5 \end{bmatrix} = \begin{bmatrix} u_{1x} & u_{1y} \\ u_{2x} & u_{2y} \\ u_{3x} & u_{3y} \\ u_{4x} & u_{4y} \\ u_{5x} & u_{5y} \end{bmatrix} \tag{2.2}$$

In order to create the nominal controller, the Laplacian of the given configuration is needed, in this case, the Laplacian matrix is:

$$L(G) = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 4 & -1 & -1 & -1 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \begin{cases} \text{if } i = j \Rightarrow N_i \\ \text{if } i \neq j \text{ and } j \in N_i \Rightarrow -1 \\ \text{else } \Rightarrow 0 \end{cases} \tag{2.3}$$

The nominal controller for each agent $i \Rightarrow u_{\text{nom,i}} = \sum_{j \in N_i} (p_j - p_i + (p_{di} - p_{dj}))$. Where p_d is the desired position for each agent and has the same dimensions as p. By changing p_d, different formation shapes can be easily created. To arrive to the final formulation of the nominal controller used in Equation 2.1, the Laplacian previously shown is used such that:

$$u_{\text{nom}} = -L(G)p + L(G)p_d = -L(G)(p - p_d) \tag{2.4}$$

2.1.1 Inter-collision avoidance and communication maintenance

Using the system dynamics, the constraint formula can be simplified to $\Delta h^T u \geq -\alpha(h)$. The lines of the formation (that is the edges) determine which robots are the neighbours of each and the collision avoidance and communication maintenance will always be assured between robots that are neighbours. Taken into account that this is the centralized version, this means that there will be two safety constraint per edge in the formation. Therefore, the controller becomes:

$$\begin{aligned} \min_{u} \quad & ||u - u_{nom}||^2 \\ \text{s.t.} \quad & \Delta h_{CM}{}_j^T u \geq -\alpha(h_{CMj}) \\ \text{s.t.} \quad & \Delta h_{OA}{}_j^T u \geq -\alpha(h_{OAj}) \\ \text{for} \quad & j = 1, 2, ..., M \end{aligned} \tag{2.5}$$

where CM means communication maintenance, OA is obstacle avoidance and M is the number of edges in the formation. The CBF for each CM and OA should have an affine form: $Au + b \geq 0$, which in the case being considered means:

$$\begin{bmatrix} (x,y)(x,y) & 0 & 0 & 0 & \\ 0 & (x,y)(x,y) & 0 & 0 & \\ 0 & (x,y) & 0 & (x,y) & 0 \\ 0 & (x,y) & 0 & 0 & (x,y) \end{bmatrix} \begin{bmatrix} u_1^T \\ u_2^T \\ u_3^T \\ u_4^T \\ u_5^T \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \geq 0 \tag{2.6}$$

where A has a dimension: (number of constraints)x(number of agents*dimension of the problem), u has a dimension: (number of robots*dimension of the problem) and b is: (number of constraints). Since u has different dimensions than u_{nom}, the latter would have to be converted to the correct ones. Assuming we only communication maintenance is used since obstacle avoidance is done the same way, but with opposite sign:

$$\begin{gathered} h\left(p_i, p_j\right) = r^2 - \left\|p_i - p_j\right\|^2 \quad \forall (i,j) \in E \Rightarrow E = \{(1,2),(2,3),(2,4),(2,5)\} \\ \text{where } \left\|p_i - p_j\right\|^2 = \left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 \end{gathered} \tag{2.7}$$

For vector b, taking into account that $h\left(p_i, p_j\right) = h\left(p_j, p_i\right)$:

$$b = \alpha(h) = \alpha \cdot h = \alpha \cdot \begin{bmatrix} h\left(p_1, p_2\right) \\ h\left(p_2, p_3\right) \\ h\left(p_2, p_4\right) \\ h\left(p_2, p_5\right) \end{bmatrix} \tag{2.8}$$

For matrix A, the partial derivatives of the CBF function h are needed:

$$\begin{array}{l} \partial h/\partial x_i = -2\left(x_i - x_j\right) \\ \partial h/\partial y_i = -2\left(y_i - y_j\right) \\ \partial h/\partial x_j = 2\left(x_i - x_j\right) \\ \partial h/\partial y_j = 2\left(y_i - y_j\right) \end{array} \quad \Rightarrow \partial h/\partial x_i = -\partial h/\partial x_j \text{ and } \partial h/\partial y_i = -\partial h/\partial y_j \tag{2.9}$$

for edge $(1,2)$, the safety constraint is calculated as:

$$\left.\begin{array}{l} h\left(p_1, p_2\right) = r^2 - \left\|p_1 - p_2\right\|^2 \\ \partial h/\partial x_i = -2\left(x_1 - x_2\right) \\ \partial h/\partial y_i = -2\left(y_1 - y_2\right) \\ \partial h/\partial x_j = 2\left(x_2 - x_1\right) \\ \partial h/\partial y_j = 2\left(y_2 - y_1\right) \end{array}\right\} \quad \text{CBF constraint for the edge:} \quad \nabla h^\top u + \alpha(h) \geq 0$$

$$\left[\partial h/\partial x_i \quad \partial h/\partial y_i \quad \partial h/\partial x_j \quad \partial h/\partial y_j\right] \begin{bmatrix} u_1^\top \\ u_2^\top \end{bmatrix} + \alpha\left(h\left(p_1, p_2\right)\right) \geq 0$$

$$-2\begin{bmatrix} x_1 - x_2 & y_1 - y_2 & x_2 - x_1 & y_2 - y_1 \end{bmatrix} \begin{bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \end{bmatrix} + \alpha\left(r^2 - \left\|p_1 - p_2\right\|^2\right) \geq 0$$

extrapolating this for all the edges, the final matrix A is obtained as:

$$A = \begin{bmatrix} \frac{\partial h_{12}}{\partial x_2} & \frac{\partial h_{12}}{\partial y_1} & -\frac{\partial h_{12}}{\partial x_1} & -\frac{\partial h_{12}}{\partial y_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial h_{23}}{\partial x_2} & \frac{\partial h_{23}}{\partial y_2} & -\frac{\partial h_{23}}{\partial x_2} & -\frac{\partial h_{23}}{\partial y_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial h_{24}}{\partial x_2} & \frac{\partial h_{24}}{\partial y_2} & 0 & 0 & -\frac{\partial h_{24}}{\partial x_2} & -\frac{\partial h_{24}}{\partial y_2} & 0 & 0 \\ 0 & 0 & \frac{\partial h_{25}}{\partial x_2} & \frac{\partial h_{25}}{\partial y_2} & 0 & 0 & 0 & 0 & -\frac{\partial h_{25}}{\partial x_2} & -\frac{\partial h_{25}}{\partial y_2} \end{bmatrix} \tag{2.10}$$

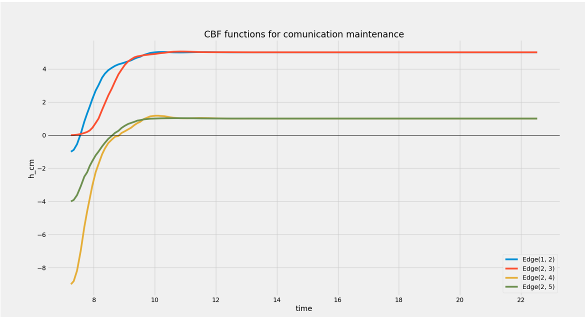
To add a simple HuIL element to the problem, for one of the robots (in this case robot 5) the output of the QP (Quadratic Program) controller will be ignored and instead a linear filter using the nominal formation controller and a human input u_{HuIL} will be used, such that: $u_5 = \gamma u_{HuIL} + (1-\gamma) u_{nom,5}$, where $\gamma = 0.5$ for the simulations and testing.

All of this, has been coded in a modular way such that the number of robots, formation shape and neighbour configuration can be modify at will and the controller variables and QP problem will be updated accordingly.

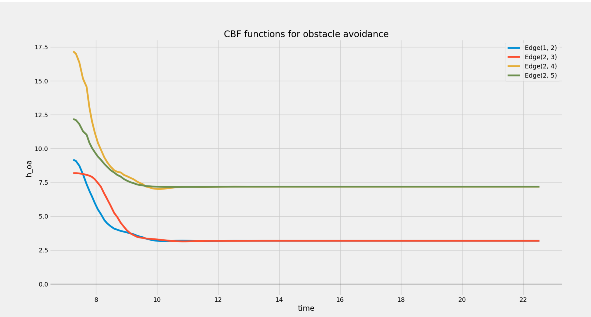
2.1.2 Simulated results

Four different cases where used to simulate this first algorithm, those were:

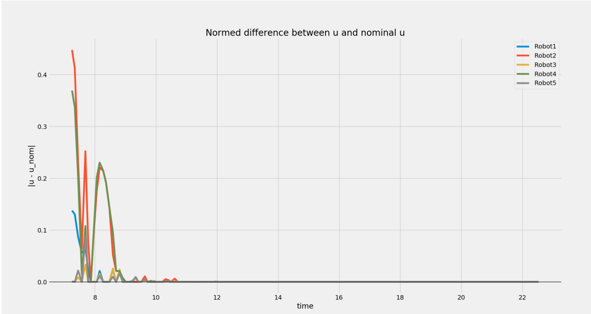
- A normal K-shape formation with a wide enough CM and OA constraints to not modify the formation controller.



(a) CBF CM



(b) CBF OA

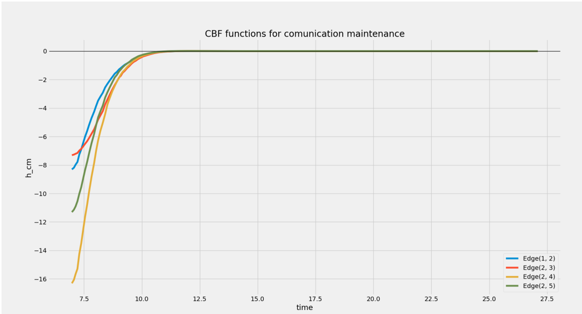


(c) Normed difference

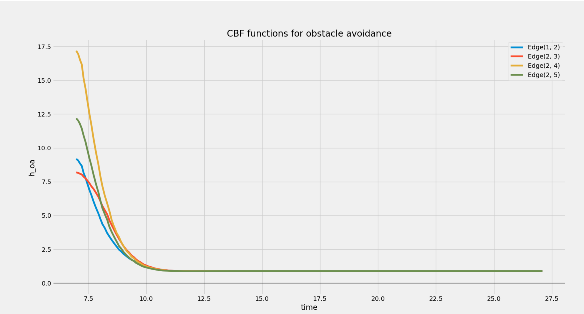
Figure 2.2: Evolution of the CBF functions and normed difference for K with CM and OA

As expected, both h_{CM} and h_{OA} are all greater than zero once the initial positions have been corrected. The normed difference between the nominal and final controller also converges to zero due to the constraints being satisfied by the nominal controller alone.

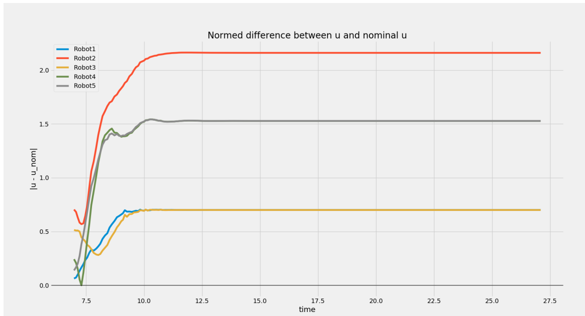
- A constrained K-shape formation by using a smaller CM and OA constraints that will force the robots to converge to a smaller shape.



(a) CBF CM



(b) CBF OA

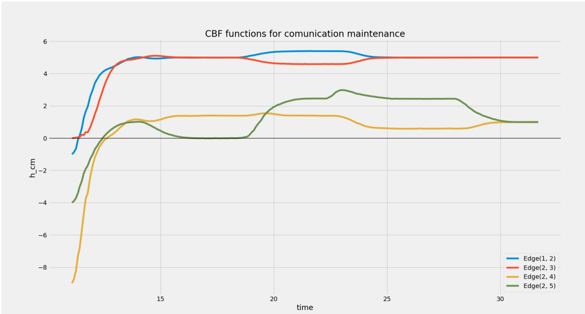


(c) Normed difference

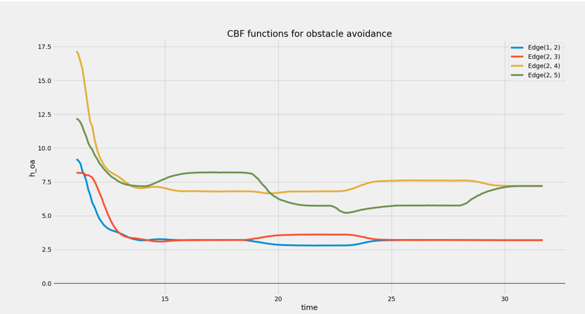
Figure 2.3: Evolution of the CBF functions and normed difference for K-constrained with CM and OA

In this case, since the nominal controller alone does not satisfied the safety constraint, the final controller will always have a positive normed difference with the nominal one, as long as the CBF functions are non-negative.

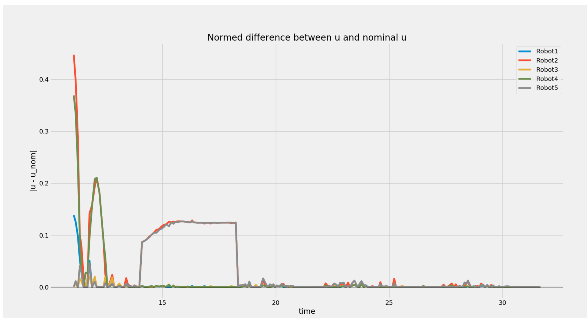
- The normal K with a HuIL robot with and without CM and OA constraints.



(a) CBF CM



(b) CBF OA

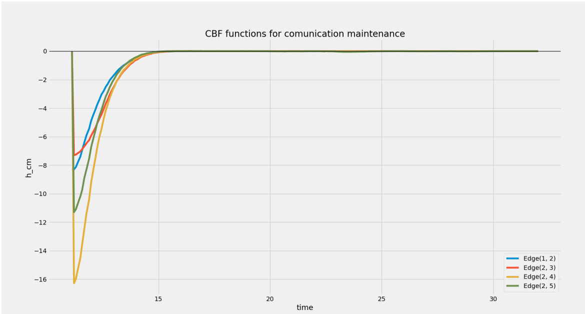


(c) Normed difference

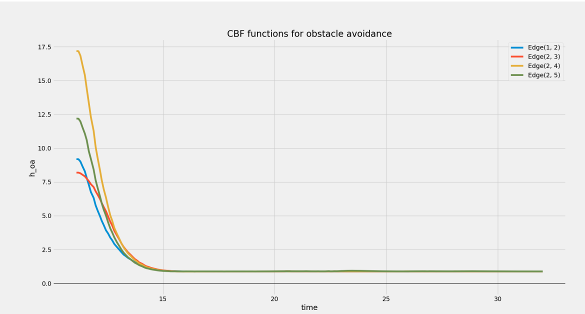
Figure 2.4: Evolution of the CBF functions and normed difference for K with HuIL and with CM and OA

Only the case with the constraints is shown since the one without it was only use as a baseline for comparison. The main difference it has with the previous ones is that there are spikes and sections in the CBF functions were it goes closer to zero, but it never crosses it since the constraint has to always be satisfied. The normed difference plots shows the main region where the HuIL was non-zero with the step and the latter smaller non-zero human commands of smaller time span.

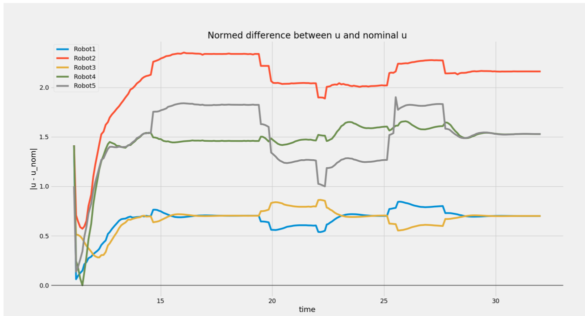
- A constrained K with a HuIL robot with smaller CM and OA constraints.



(a) CBF CM



(b) CBF OA



(c) Normed difference

Figure 2.5: Evolution of the CBF functions and normed difference for K-constrained with HuIL and with CM and OA

The same thing happens as before, but the normed difference never goes to zero as it was explained in the previous cases.

Videos about each of these cases can be found on the documentation folder of the Thesis GitHub.

2.1.3 Arena safety constraint

The limits of the arena at the SML are: $x_{max} = 2m$, $x_{min} = -1.4m$, $y_{max} = 1.8m$ and $y_{min} = -3m$. This means that for each robot there will be 4 safety constraints represented for the CBF functions as:

$$h^i_{x\,\max} = x_{\max} - x_i$$
$$h^i_{x\,\min} = x_i - x_{\min}$$

$$h^i_{y_{\max}} = y_{\max} - y_i$$
$$h^i_{y_{\max}} = y_i - y_{\min}$$

For each agent, the CBF constraint is $\Delta h_i^T u_i + \alpha(h_i) \geq 0$, where: $\nabla h = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$ and $b = \alpha(h_i) = \alpha \begin{bmatrix} x_{\max} - x_i \\ x_i - x_{\min} \\ y_{\max} - y_i \\ y_i - y_{\min} \end{bmatrix}$.

For the centralized problem assuming 3 agents, the constraints can be joined together in the form of

$Au + b \geq 0$, where the A matrix and u controller are:

$$u = \begin{bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \\ u_{3x} \\ u_{3y} \end{bmatrix} \Rightarrow \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The vector b is just an extension of the one shown for individual agents and therefore it is skipped.

2.1.4 Wedge shape safety constraint

This constraint is similar to the one used for the arena, but only two constraints are needed to make sure that the robot does not go outside of the following wedge shape (defined using the available space at the SML):

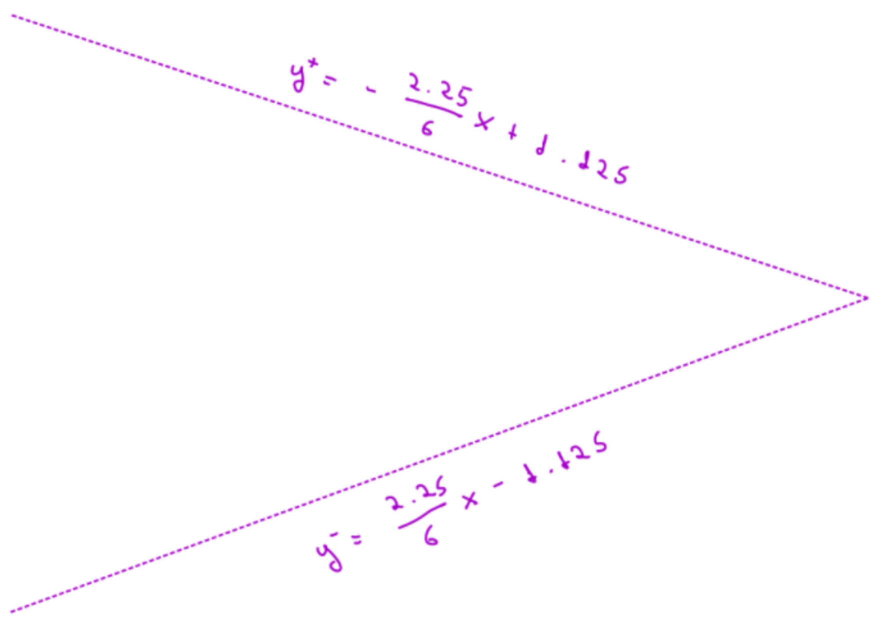


Figure 2.6: Wedge shape for the SML

The CBF functions for each agent are:

$$
\begin{aligned}
h^i_{\max} &= y^+ - y_i = -\frac{2 \cdot 25}{6} x_i + 1 \cdot 125 - y_i \\
h^i_{\min} &= y_i - y^- = y_i - \frac{2 \cdot 25}{6} x_i + 1.125
\end{aligned} \tag{2.11}
$$

so that $\nabla h = \begin{bmatrix} -\frac{2.25}{6} & -\frac{2.25}{6} \\ -1 & 1 \end{bmatrix}$ and $b = \alpha \begin{bmatrix} y^+ - y_i \\ y_i - y^- \end{bmatrix}$.

For the centralized problem assuming 3 agents this becomes:

$$
A = \begin{bmatrix}
-\frac{2.25}{6} & -1 & 0 & 0 & 0 & 0 \\
-\frac{2.25}{6} & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{2.25}{6} & -1 & 0 & 0 \\
0 & 0 & -\frac{2.25}{6} & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{2.25}{6} & -1 \\
0 & 0 & 0 & 0 & -\frac{2.25}{6} & 1
\end{bmatrix} \tag{2.12}
$$

With b being an extension of the one shown for one agent and u being the same as for the wedge case.

2.1.5 Extra Human-In-The-Loop agent

In this subsection a new CBF condition will be introduced with an extra robot that is fully controlled by a human (or maybe it is a human itself) and it is not taking into account for the QP. Each of the robots in the formation would have to make sure they have enough distance with this extra robot to avoid collision. For one agent this becomes:

$$
\begin{array}{ll}
\text{For the agent:} & \text{For extra-HuIL:} \\
u_i = \begin{bmatrix} u_{ix} \\ u_{iy} \end{bmatrix} & p_e = [x_e, y_e]
\end{array} \tag{2.13}
$$

$$
\begin{aligned}
& h(p_i, p_e) = \|p_i - p_e\|^2 - r^2 \\
& \text{ where } \|p_i - p_e\|^2 = (x_i - x_e)^2 + (y_i - y_e)^2 \\
& \nabla h(p_i, p_e) = \begin{bmatrix} \partial h / \partial x_i \\ \partial h / \partial y_i \end{bmatrix} = \begin{bmatrix} 2(x_i - x_e) \\ 2(y_i - y_e) \end{bmatrix} = 2 \begin{bmatrix} x_i - x_e \\ y_i - y_e \end{bmatrix} \\
& A = 2 [x_i - x_e \quad y_i - y_e] \\
& b = \alpha(h) = \alpha \cdot h = \alpha \cdot \left(\|p_i - p_e\|^2 - r^2 \right)
\end{aligned}
$$

For the centralized problem assuming 3 agents the constraint can be expressed as $Au + b \geq 0$, where:

$$
u = \begin{bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \\ u_{3x} \\ u_{3y} \end{bmatrix} \quad
A = \begin{bmatrix}
2(x_1 - x_e) & 2(y_1 - y_e) & 0 & 0 & 0 & 0 \\
0 & 0 & 2(x_2 - x_e) & 2(y_2 - y_e) & 0 & 0 \\
0 & 0 & 0 & 0 & 2(x_3 - x_e) & 2(y_3 - y_e)
\end{bmatrix}
$$

$$
b = \alpha \begin{bmatrix} \|p_1 - pe\|^2 - r^2 \\ \|p_2 - p_e\|^2 - r^2 \\ \|p_3 - pe\|^2 - r^2 \end{bmatrix}
$$

2.1.6 Testing results

Three different tests have been created and performed at the time of writing this document. All of them have the safety constraint for the arena size and CM and OA for the robots inside the formation.

- 3 agents in a K formation with one being HuIL.

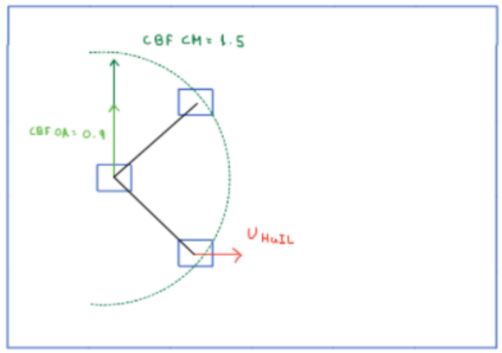
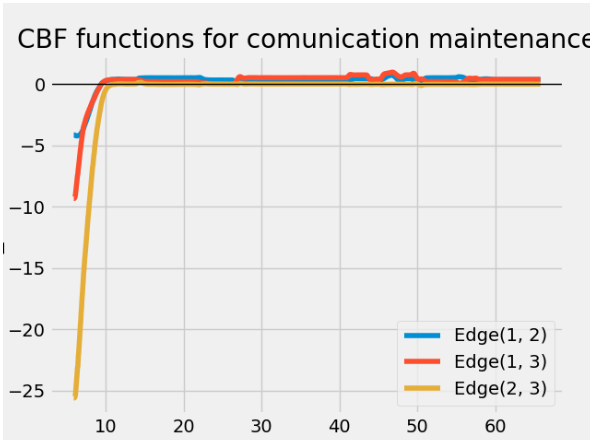
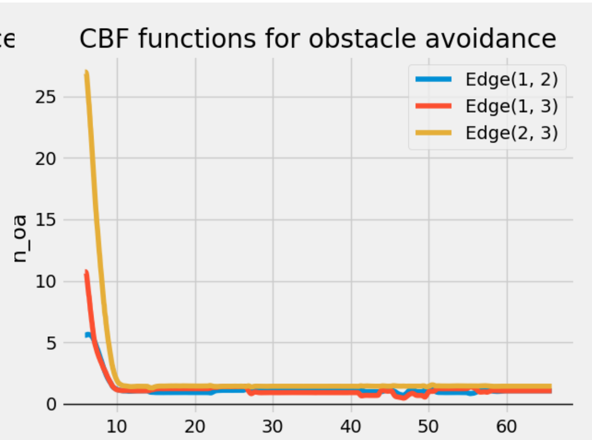


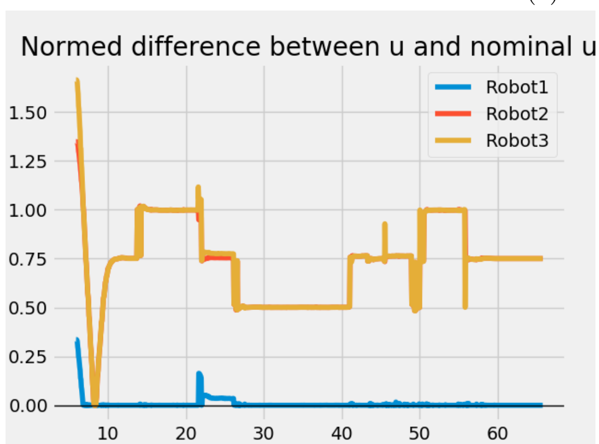
Figure 2.7: K3 HuIL



(a) CBF CM



(b) CBF OA



(c) Normed difference

Figure 2.8: Evolution of the CBF functions and normed difference for K-3 with HuIL and with CM and OA

- 3 agents in a K formation with one being HuIL and a wedge safety constraint.

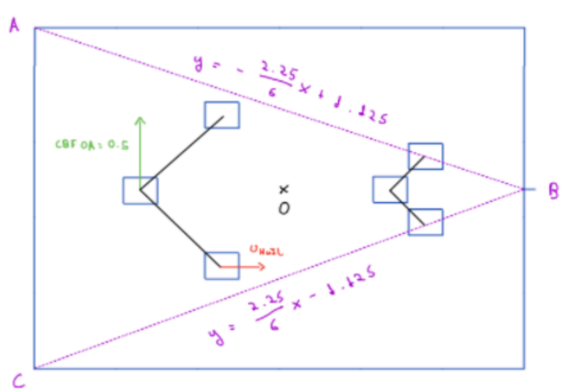
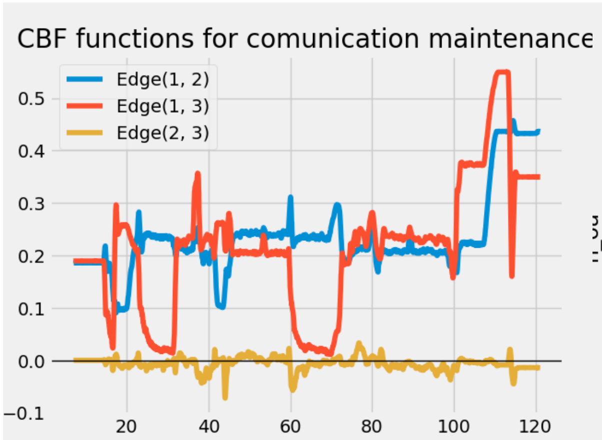
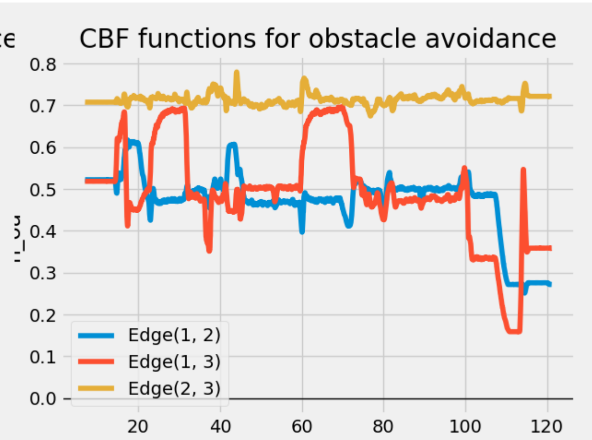


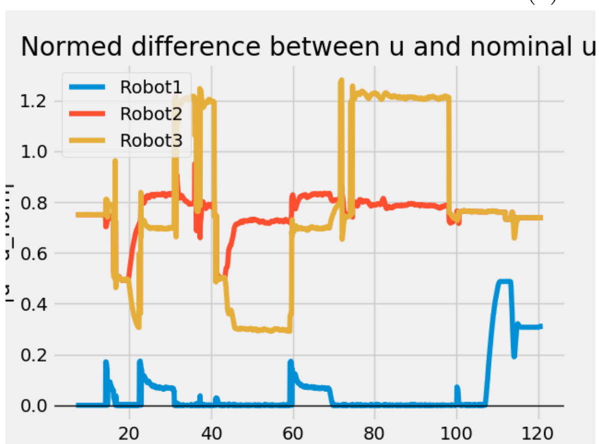
Figure 2.9: K3 wedge



(a) CBF CM



(b) CBF OA



(c) Normed difference

Figure 2.10: Evolution of the CBF functions and normed difference for K-3 with HuIL and with CM and OA and wedge

- 3 agents in a K formation with an extra HuIL robot.
  There are no plots for this.

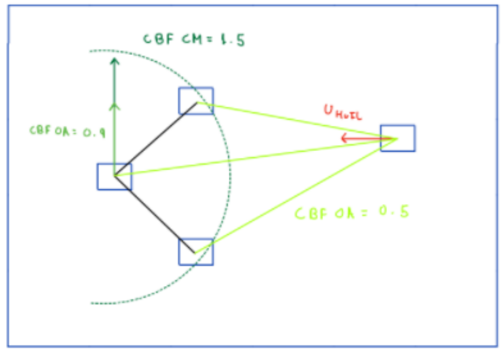


Figure 2.11: K3 Extra agent

Videos of this tests can be found in the documentation folder of the Thesis GitHub.

2.2 Dual variables decentralized version

2.3 ADMM-based decentralized version

2.4 Developed decentralized version

Extra: Satellite constellation

3.1 Background

3.1.1 Dynamics of the system

3.1.2 CBF safety constraint

3.2 2D problem

References