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

Victor Nan Fernandez-Ayala
19980323-2637
vnfa@kth.se

November 8, 2021

Contents

1 Formation with CM and/or OA

In this first section, a multi-robot formation keeping algorithm will be created with as many robots as wanted (currently max. of 5 to ensure convergence) to form whatever formation with whatever 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 enough distance between each robot and its neighbours to prevent possible collisions. Additionally, one, or several, of the robots will include a Human-In-The-Loop controlling it at an specific time frame. Done both in a centralized and decentralized way (without assured convergence for now).

1.1 Decentralized simulation

The formation controller will create a K shape with 5 different robots and the second one being in the middle, as it can be seen in the following figure. [IMAGE] The idea is to combine this with a CBF function that will force the robots to shrink the formation to a smaller K while avoiding collision as shown in the next figure. [IMAGE] For the CBF part, both for communication maintenance and obstacle avoidance, several constraints equations have been tried:

- The first one used was obtained from the following paper [TO FILL IN]:

$$(\sum_{l \in L} e^{-ph_l} h_l)(f + gu) + \alpha \sum_{l \in L} (e^{-ph_l} h_l) \geq 0 \tag{1}$$

  This has been proven to not be true, due to some mistakes in the derivation. Therefore, a previous equation was tried.

- The second one used was also derived from the same paper. It has the following form:

$$\frac{1}{\sum_{l \in L} e^{-ph_l}} (\sum_{l \in L} e^{-ph_l} h_l)(f + gu) - \frac{\alpha}{p} \log(\sum_{l \in L} e^{-ph_l}) \geq 0 \tag{2}$$

  Although it is theoretically correct, the use of CBF functions inside of an exponential meant that the result increased dramatically fast, going to numbers that ROS could not simulate (e.g. speeds around e^{-259}) whenever the robots where close or outside the safe region (which can happen in the current decentralized algorithm used). For this reason, although it is very easy to prove there is a feasible optimal solution to the minimization problem a new equation was needed.

- The final and the current equation being used is the following:

$$A \cdot u + b \geq 0 \tag{3}$$

  Where A is a matrix of dimension mxn, where m is number of neighbours for each robot and n is the dimension of the problem (in this case 2 since it is planar motion) and b is a vector of mx1 dimensions, that is each row represents a different

CBF condition associated with one of the neighbours. The main drawback of this approach is that proving that there is a feasible optimal solution is harder than before due to the extra constraints.

With this last equation results are very promising even though convergence is not assured due to the decentralized implementation. When using the first and second equation, sometimes the robots could be seen diverging due the large speeds calculated as explained before. But also, even in the best scenario where they converged to the shrinked K, the robots all ended up moving in the same direction with a small speed which was undesirable. This problem does not appear when using the current equation. The following plots show the evolution of the controller u compared with the nominal one u_{nom}, and the CBF functions, h, for each of the robots. [IMAGE] Both CM and OA are activated with a safety distance of 1.3 and 0.9, respectively. The ideal formation makes the robot be at least 2 m away between each other so this minimum distance for CM will be ideal to see if the CBF is being activated. OA is also a great addition since it prevents a lot of possible collisions when forcing the robots to be so close between each other.
One can see in the evolution of the controller that it starts almost being equal to the nominal one at the start when both the CBF constraints and the nominal controller point in the same directions and it starts diverging once the constraints force the controller to avoid the nominal controller, proving that the CBF constraints are more important for the algorithm as it was desired.

Additionally, videos of this simulation can be found in the video folder.

1.2 Centralized simulation

The parameters, equations and formation for this version are the same as before. The major difference however is that before there was one node per robot, requesting position data from the motion capture system for its position and its neighbours, calculating the controller and then inputting the speed; while now there is only one node doing all of it. This means that before there was a delay between inputting the velocity command for the first and the last robot while now that delay is negligible.
This can be shown by the fact for the decentralized case a limit of 5 robots was discovered (sometimes 6 works). If more robots were used, due to the limitations of the computer used for the calculations, the delay between the speed commands caused divergence. For the centralized case, since this delay is negligible a much higher number of robots might be used (video where I test algorithm with 7 robots in the video folder).
The following plots show the evolution of the controller u compared with the nominal one u_{nom}, and the CBF functions, h, for each of the robots as done before. [IMAGE]