ECE6563 Networked Control Systems Final Project

- ## Motivation behind your selection of project.

  The Matlab simulation consists of six different tasks, all of which are of great help to understand how networked agents work and the people manage to successfully control them in a desired way. I don't have networked control related research, but six would definitely broaden my vision in this field.

- ## A description of the controller(s).

  My controllers are pretty simple ones. Basically two types of controller are used in this project. One is responsible to keep distance between two agents while the other keeps distance and connectedness between agents at the same time.

$$\frac{k(d - d_{desired})}{(d - d_{min})^2}$$

The formula above encodes the input we give each agents, where $d_{desired}$ is the desired distance between two agents and d_{min} is the minimum distance between two agents. This formula ensures that two agents not colliding with each other and also tries to maintain the desired distance regardless of other forces from others.

For the first waypoint, I made the agents a complete graph, so that at the beginning of the second waypoint, I can easily connect the agents as a line graph using their uids. The second and third waypoint are all done by single leader and followers strategy and a line graph did a great job. At the fourth waypoint, I search the area by driving the leader to each corner and try to connect all agents in the meantime. This can be seen as a net, and the closer the agents are between each other, the finer the net and more likely that we are able to find our lost agents.

For the fifth one, I clear it by making those have small distance from the obstacle leaders so that they don’t pay attention to others. Leaders has to do one thing and one thing only, to follow the obstacle at a constant speed. Those who are not leaders, i.e. followers, need to follow the leaders with the formula above.

For the sixth one, I treated each neighbor as a push and each beacon as a pull. Therefore, when an agents stand right on the target location, the push and pull cancel each other. All I feel are those target locations without an agent standing on it. Every time an agent finds a close target, it stays there and remain moved. Others continue looking for slots until all agents are finished. This strategy works when you carefully tune the parameters.

- **Observations on the performance of your controllers.**

The performance from the first four waypoints are great in terms of speed and smoothness. However, for the fifth task, some manual tunning is needed and are very time-consuming. I watched the video from the previous students and found their groups are more compact, even for the one without the leader. The sixth one works but is comparatively slow. Since I fixed the agents once they are at the right point, other cannot pass through them. Instead, they need to move in circles to approach their target. Sometimes, if not often, the last one would find itself in a dilemma, stuck in the opposite side of the target it should be at.

I didn’t come up with better solution to the last two waypoints and the total time to finish all tasks is about two minutes.

Yujia Liu

903070716

yliu732