Project Report of Swarm Robotics

Saitong Zhao - sz2641

December 19, 2019

1 General description

In this project, I am going to collaboratively control 6 robots with two wheels to do the tasks. The tasks contains formation control, translate the swarm and use the swarm to move objects to specific places. The video is uploaded to Youtube with the link:

https://youtu.be/57zwhvkJIPQ

All the controller in the robots use local and distributed control laws. The communication range between each two robots is 2 meters and the information I used to control is the coordinate of a robot and its relative distance between its neighbors. In this implementation, every task is divided into simpler tasks, and different tasks are defined by a task state. To synchronize the specific task every robots focus on, a protocol similar to 2 Phases Consensus is designed to achieve the distribute synchronization on task id.

Consensus on task Because the robots are working individually without a centralized control method, there could be problem that the synchronization on task the swarm is working on failed. In which case the robots in the swarm working on different tasks. So I designed an algorithm with two phases to achieve the synchronization.

In the First phase: every robot periodically detecting whether the task it is currently working on is finished or not. If this robot recognize this task as finished, it will update its own attribute of *next task*. When this robot communicating with its neighbors, this attribute of *next task* will be appended in the message. In the Second phase: on every robot, after received the messages from the neighbors, the robot will compare its own *next task* value with the according value it received in the messages. If all the *next task* value is the same, the robot will regard this as a consensus and set the task it is working on as the value in the *next task*.

This algorithm can make sure that the swarm is working on the same task. Since this is a simpl simulation, this algorithm can achieve this job. If we want to make this swarm system more robust, we could use the concept of voting in PAXOS protocol to enhance this consensus algorithm.

Collision Avoidance In the experiment, though the robots can have some parts overlapped, but it is still bad for two different robots to be too close. For some control law, two robots in a very short distance might cause big change on velocity and may lead to a turn over of a robot.

So here I used an artificial potential field to achieve a collision avoidance. Different from the potential we encountered on the class, I only use part of it to simulate a repulsive force field. The closer two robots is, the magnitude of the force is higher. The function of repulsive force field is:

$$V_R(d_{ij}) = \begin{cases} \dfrac{1}{d_{ij}+1}, d_{ij} < d_1 \\ 0, d_{ij} >= d_1 \end{cases}$$

By adjusting the parameter d_1, the collision avoidance can be achieved.

Communication Range Control As learnt on class, we can use weighted feedback control combined with other control laws to achieve control on robots in case they go too far away and exceed the communication range. But here in some steps I designed, the distance between some specific robots have to be more than 2 meters. And is the experiment, this out-of-communication error don't exist, so the weighted feedback control is not implemented.

2 Task-Step Detail and implentmentation

Task 1 - square formation This is a simple task can be achieved within only one step. In my implementation, I use relative state-based control law, for the robots to form a square with length of side equal to 1. The control law is:

$$u_i = -K \sum_{j \in N_i} (x_i(t) - x_j(t)) - (p_i - p_j)$$

The simulation result is displayed in Figure 1.

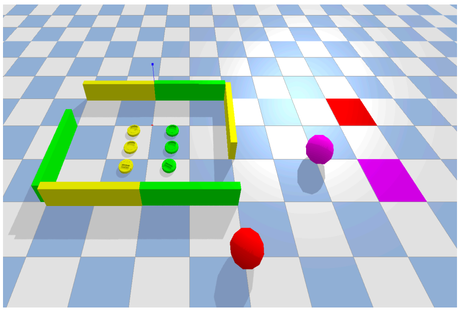

Figure 1: Square formation

Task 2 - Get out of the room and circle formation I divided this task into two steps. First get the swarm out of the room and then maintain a formation of a circle.

To move the swarm out of the room, I modified the topology as a directed graph with a root robot as leader. Every robot is only connected with one other robot. This is because of the door of the room is very small, and the robot can only come out one by one. Agreement with a leader can achieve this on the topology as a linked list. Simulation result of this step is displayed in Figure 2

$$\dot{x} = -Lx$$

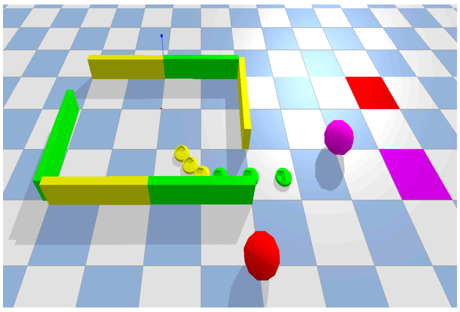

Figure 2: Line out

The translation step is considered to end when the last robot get out of the room and is 0.5 meters away from the boundary. Then I would use potential field with a virtual leader to make a circle formation with the swarm. Because I would use caging to move the purple ball in the next task, here I make the purple ball as the center of the circle.

To form a circle, I use a virtual leader's potential field, with the potential field function introduced on the class. The shape of the points with lowest potential energy combined together, is a circule. And the parameter d_0 can be used to define the radius of the circle, parameter d_1 can be used to define the size of the field, parameter α can be used to define the magnitude of the force given by the field.Then, to make the robots evenly distributed on the edge of the circle, the repulsive force field is also used. It is needed to mention that, the effection range of the repulsive field should be larger than the radius of circle. And the α parameter of both fields should be properly designed and tested, in case each of them is too large and break the control we would like to do. The potential function is:

$$V_h(d_{ik}) = \begin{cases} \alpha_h(ln(d_{ik}) + \dfrac{d_0}{d_{ik}}), 0 < d_{ik} < d_1 \\ \alpha_h(ln(d_1) + \dfrac{d_0}{d_1}), d_{ik} > d_1 \end{cases}$$

And the control law of each robot is:

$$u_i = -\sum_{j \neq i}^{n} \nabla_{r_i} V_R(d_{ij}) - \sum_{k=0}^{m-1} \nabla_{r_i} V_h(d_{ik}) + f_{v_i}$$

Task 3 - Move the purple ball to the purple floor In this task, I used caging method to move the ball. Because in this simulation environment, the friction force is weak, and the robots don't have sensors to locate the ball and navigating to the destination, push method will be unstable and time consuming. The robots also don't have machine arm, so the grasping method is also invalid.

So for this task, the first step is use the swarm to force a cage, which is implemented as a change on the radius of virtual leader's potential field. When the cage is created, we can move the purple ball by moving the center of virtual leader's potential field. The speed should be neighter too high not too low, or the formation might be destroyed. Since I know little about the physic parameter of the simulation world I tested to find possible value of swarm speed to move the ball. The simulation result of this task is displayed in figure 3:

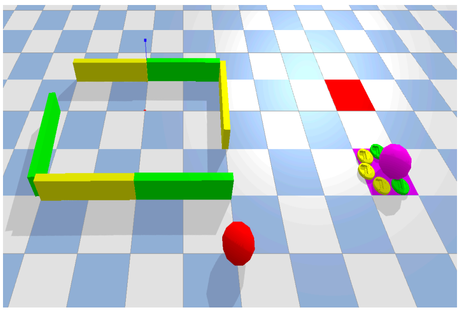

Figure 3: Move purple

Task 4 - Move the red ball to the red floor For this task, the robots need to first deform the cage and get away from the purple, move to the place near the red ball, form a cage surrounding the red ball and move the red ball to the red floor with this cage.

So first, I increase the radius of virtual leader potential field, to make the distance from the robots to the purple ball further. Then I delete the virtual leader potential field and use the simialr method in Task 2, move the swarm to a position near the red ball with a rooted agreement protocol. The simulation run as displayed in figure 4.

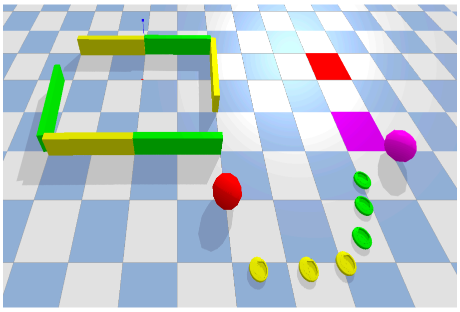

Figure 4: Head to red

After the swarm has moved to the position near the red ball, a new artificial potential field of virtual leader is activated, guiding the robots form a circle around the red ball. Then, similar to the Task 3, the virtual leader move to the destination, and the swarm will carry the red ball to the destination. Result in figure 5.

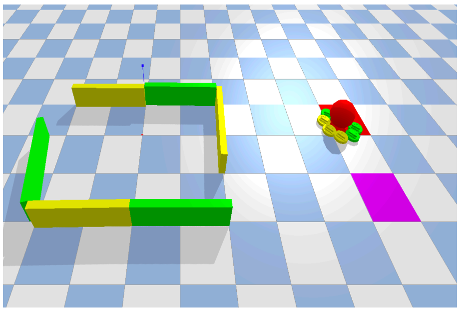

Figure 5: Move Red

Task 5 - Go back to the room and diamond formation Generally, this task is simialr to Task 1 and Task 2. All I did is in the same way as before. Using a new topology with a leader, force the robots go in a line, get back into the room. And then using the relative state-based control to form a diamond formation. And typically, a diamond formation would only need 4 or 8 robots. So here the diamond is not looks as comfortable as a 4-robot or 8-robot one.