Decentralized control and path planning for navigation and manipulation of multi-agent system in a known environment

Rajath Koratagere Manjunath
N19536684

No Institute Given

Abstract. We present a framework for decentralized control of multi-agent system in a known environment for with various obstacles and constraints. Specifically, the multi-agent system consisting of six mobile robots has to initially form a square in a given area. Next, the system has to navigate the agents out of the room and form a circle. Then the agents manipulate the balls of purple and red color to squares of the same color and navigate back into the room make a diamond shape.

1 Introduction

Control of multi-agent system is an exploring problem due to its practical potential and theoretical challenges in their control and coordination. Major challenge is making decisions about the system using partial information available to the agents.This has to be done without the intervention of a central controller. Another problem is to retain particular shape in space while navigating from one place to another due to the application restriction.

The work here aims to control a fleet of six robots and perform a set of tasks in the known environment and navigating them by avoiding obstacles. The set of tasks are performed in the following sequence:

- A square is made using the six robot.
- The agents are navigated outside the box avoiding collision with obstacles and other agents.
- A purple ball in the environment is moved to a purple ball.
- A Red ball in the environment is moved to a red ball.
- The agents get back into the box without colliding with other robots or the obstacles.
- They form a diamond inside the box.

In the next section, we'll describe our problem with more technical details and explain in detail our methodology.

2 Methodology

In the subsection 2.1, we'll briefly explain control methods for arranging the six robots in a square, a circle and a diamond. In subsection 2.2, we'll briefly describe the method for navigating the multi-agent robot system in the environment. In the subsection 2.3, we'll describe the potential function for obstacle avoidance and avoiding collision with other agents.

2.1 Formation control

A multi-agent system is a framework consisting of graph and a function mapping the vertices to the points in space for a given formation. The graph consists of vertices and edges. The function maps the graphs to a point in d dimension. The function can be used to specify the constraints between vertices.
We use rigidity matrix based formation control for both square and triangle formation using the six robots. The multi-agent framework consists of robot as vertices, communication between them as edges and their location in the environment as p. We use the graph which describes the shape to describe our control law. The control law is given by

$$\dot{p} = k((p - \bar{p} - (p_{des} - \bar{p_{des}})) \tag{1}$$

where p are the current positions, and p_{des} are the desired positions
The advantage of this control law is it is rotational variant and its con is that it does not use the concept of distance and hence less easy to calculate. The desired position of the vertices for square formation are:

$$\{v_0, v_1, v_2, v_3, v_4, v_5\} = \{1., 1.0, 1., -1.0, 0., 1., 0., -1., -1., 1.0, -1., -1.0\}$$

The desired position of the vertices for circle formation are:

$$\{v_0, v_1, v_2, v_3, v_4, v_5\} = \{\cos\pi/3, \sin\pi/3, 1, 0, -\cos\pi/3, \sin\pi/3, \cos\pi/3, -\sin\pi/3, -1, 0, -\cos\pi/3, -\sin\pi/3\}$$

The desired position of the vertices for diamond formation are:

$$\{v_0, v_1, v_2, v_3, v_4, v_5\} = \{1., 1.0, .5., -\cos(\pi/6), 0., 1., -.5., -\cos(\pi/6), -1., 1.0, -1.5, \cos(\pi/6)\}$$

2.2 Path Planning

We want to move towards a particular point. Hence we try to minimize the distance between the position of the agents and the desired point. The control law is given by

$$\dot{p} = k(p - p_{des})$$

2.3 obstacle avoidance

In the known environment, we know where the obstacles are, hence we consider points in obstacle and try to maintain a certain distance in between them. The control law used for this is given by

$$\dot{p} = \begin{cases} sgn(p - obs) * e^{|p-obs|}, & \text{if } d \leq d_{thres} \\ 0, & \text{otherwise} \end{cases}$$

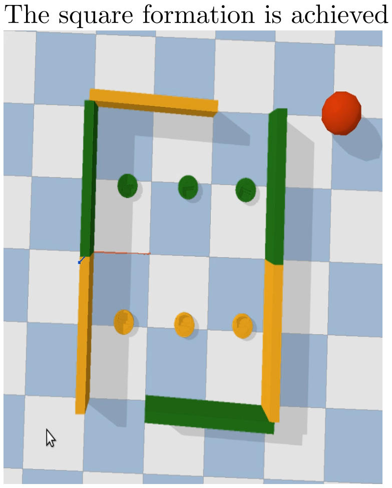
A similar control law is used to maintain a minimum distance between agents. It is given by

$$\dot{p} = \begin{cases} sgn(p - \bar{p}) * e^{|p-\bar{p}|}, & \text{if } d \leq d_{thres} \\ 0, & \text{otherwise} \end{cases}$$

3 Results

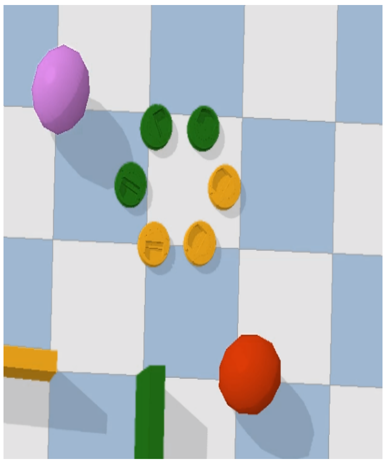
3.1 Square Formation

The square formation is achieved using the formation control



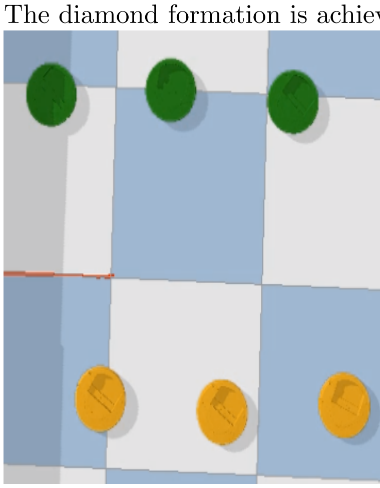
3.2 Circle formation

The circle formation is achieved using the formation control



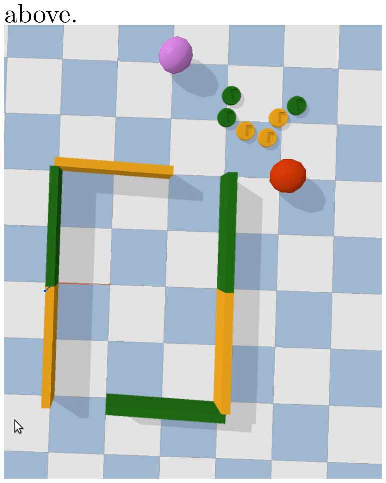
3.3 Diamond formation

The diamond formation is achieved using the formation control



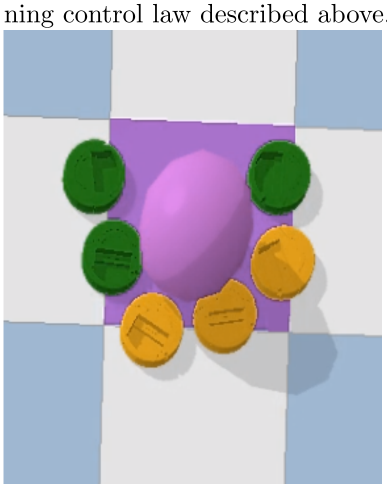
3.4 Navigating out of the room

The agents navigate out of the box using the path planning control law described above.



3.5 Placing purple ball on the purple square

The purple ball is placed on the purple square using a combination of the above explained control laws. The agents navigate out of the box using the path planning control law described above.



3.6 Placing red ball on the red square

The red ball is placed on the red square using a combination of the above explained control laws. The agents navigate out of the box using the path planning control law described above.