Lighting Talk

Script

Slide 1

Problem

Problem:

- The coordination of multiple distributed behaviors across a network with more than one decision maker.

Aim (Simplification):

- Make a team of robots assemble into a desired geometric formation.

Solution:

- To address such a problem or more-so goal, of operation within coordinating and controlling multiple robots, engineering methods revolving around *formation control* must be employed.
- More specifically within the realm of *formation control*, there are two main methods I will attempt to discuss or explore:
    - Reaching Decentralized Agreement
    - Weighted Protocols

Introduction

- To have robots come together and solve team-level, global tasks (meaning on an open environment) we need to employ *local* interaction rules.
    - Local is used in the sense that individual robots can only act on information that it has available to it – i.e. through sensing or active communications. This may also be called *distributed*.
- To achieve *locality,* we want to make sure the algorithms we use are indeed *local*. The first aspect of *formation control* being *reaching a decentralized agreement,* therefore, involves employing such algorithms based solely on the measurable information available to the robots.
- Therefore, these equations include the *relative displacements* that are employed within the equations in such a way that the robots can achieve a *global* objective.
- Hypothetically, let's say we extend this problem and we want the robots to meet at a common location – the robots do not necessarily know where they are so *a priori* agreed meeting location can be used.
- This can be accomplished by using what is known as the *rendezvous problem* – a method to evaluate how well a robot is performing in travelling to its aim, by evaluating the total error of this performance.
- Within the equation, the gradient of the total error, with respect to the individual robot positions can be determined – using the gradient descent flow allows a direct method to minimize such an error function. Therefore, this equation lets us comprehend the dynamics of the robot such that the robots move in the direction of the negative gradient of the total error.

- This is referred to as the node-level dynamics of the system since it describes the movements of the individual robots.
- To analyse the behavior of the global system, we need to ensemble-level dynamics, noting that the node-level dynamics is linear.
- Furthermore, this equation is simplified to utilize relative position differences between adjacent robots. We can express the previous node-level dynamics with an ensemble form shown.

Reaching Decentralized Agreement

- The *rendezvous problem* is beneficial for three reasons as it hints at a systematic method of obtaining decentralized multi-robot control laws by starting with an error function and then producing robot motions that explicitly reduce this error.
- It calls out the sometimes-intricate coupling between robot motions and the evolution of the underlying network structure. Simply, what makes distributed multi-robot control tricky is that it is not enough to consider the individual motions.
- The motions must be understood in conjunction with their effects on the underlying network structure. The *node-level dynamics equation* by itself is the key to addressing this problem, is also known as the *consensus equation*.
- The reason for this is because moving around, the robots are agreeing (or reaching a consensus) on where they want to meet. This is visually displayed in the first figure on the left as an example of its performance when applied to robotic dynamics.
- Instead of reaching agreements over the positions, the robots agree on what direction they should move in – *consensus equation* operates on the robot headings instead of their positions. This results in a *flocking* behavior, as illustrated on the right with the change of headings to achieve a common goal.
- Since the equation only involves measurable information of $x_i - x_j$ – scalable. Furthermore, the equation only involves the neighborhood, as opposed to the full set of robots.
- Therefore, it will indeed drive all robots to a common position if the underlying information exchange network is "rich enough".

Term Definitions

- $x_i - x_j$ Positions of near-by robots relative to their own positions (between two robots i and j).
- Total Error Equation – Gradient of the Total Error with respect to the individual robot positions between a constant robot j and all the other robots $i = 1, \dots, N$.
- Arrange themselves at a prescribed inter-robot distance δ, we obtain a *formation control* protocol as opposed to *rendezvous protocol*.
- Δ is the distance where the robots are no longer able to sense each other.

Slide 2

- Furthermore, manipulation of this equation by using a 2×2 *identity matrix & rotation matrix angle*, within the *node-level dynamics equation*, the robots instead let their aim be towards their neighbor in a *cyclic pursuit*. They will spiral outwards away from each other as illustrated on the right.

Limitations

- In the static and directed case, *consensus* is achieved. In the dynamic case, the conditions are slightly more involved.

- Furthermore, there is the possibility, with *rendezvous*, of collisions among robots.
- The remedy to this is to turn the *consensus equation* into a truly useful multi-robot coordination law, we need to augment it to ensure that the robots do not get too close to each other.

Weighted Protocols

- The construction of the *Reaching Decentralized Agreement* can be generalized by defining a symmetric, pairwise performance cost between robots. Therefore, defining the global performance cost.
- Uses a scalar function of the inter-robot distance times the relative displacement.
- Therefore, all algorithms within *weighted protocols* display the possibility of achieving rich and diverse multi-robot responses through a systematic selection of scalar weights in the *consensus equation*.
- The interpretation here is that, as fish pay more attention to near-by fish, the square norm counter-acts this by penalizing far-away fish in an overly aggressive manner.
- If the objective is not to assemble a particular shape, but rather to spread the robots, cover an area, modifications to this construction are needed.

Example of Applications

- **Factory Floor:** Multiple robots cooperating through deep learning and navigating to specific objectives and rendezvousing in a specific coordinate or position.
- **Natural Resource Monitoring:** Robots communicating over a network that compiles areas that have been already monitored, scanned, studied, etc.