Safely Reconfiguring Formations in Cluttered Environments

Aishwarya Unnikrishnan
Robotics Institute
University of Michigan, Ann Arbor
Email: shwarya@umich.edu

***Abstract*—This work presents introduces a cost-metric and exploration strategy for safely navigating a formation of robots in a cluttered environment when combining a high-level global planner (based on A*/RRT*) with a low-level local planner based on barrier functions that is used for control. The proposed algorithm firstly, aims to generate a collision-free trajectory of waypoints that serve as a reference for the multi-agent system. Secondly, the algorithm chooses the safest formation in the planning space to switch to according to distance from the obstacles and formational error. The agents in the system are assumed to be able to communicate their reference trajectory with each other and to possess a pre-assigned leader that broadcasts the same.**

I. INTRODUCTION

The objective of this work is to generate a reference trajectory offline or intermittently for the formation reconfiguration of a multi-agent system with one leader and any number of followers. Each agent in the multi-agent system has the following components:

1) Communication: wherein the leader broadcasts a reference trajectory message to all the followers
2) Local Controller: where each agent has a low level controller that can track a set of waypoints.
3) Safety parameters: A radius of collision avoidance r_c, a minimum safety distance d_s & a radius for control with collision avoidance to become active d_c are defined, such that $d_c > d_s \geq 2r_c$.

II. PROBLEM FORMULATION

A. Problem Statement

In this work, a leader-follower system with N agents is considered and can be represented as an undirected interaction graph G = (V,E), where V is set of vertices representing each agents' position and E is the set of edges between them that accounts for the adjacency relation between them, encoded by a proximity distance r_s.

The leader-follower system has to take arbitrary formations (or shapes). Let **f** be a formation from a set of formations F and Q = $\{1, \ldots, N\}$ be the set of all agents. Let i be the index of any agent in the formation, where i $\in Q$ such that $\tau = \{\tau_i \in R^2|_{i \in Q}\}$ is the set that encodes relative distances from an arbitrary point in the formation $(c_x, c_y) \in R^2$. Thereby, the pairwise inter-agent distances can be found by $\tau_{rel} = \{\tau_{ij} = \tau_i - \tau_j \,\forall\, (i,j) \in (Q,Q) | i \neq j\}$. Each f $\in F$ has a τ^f_{rel} associated with it. A pictorial demonstration of the same is shown in Figure 1. This work considers shapes like squares, rectangles, circles and stars so the standard deviation of the elements in τ are comparable to elements in τ_{rel}.

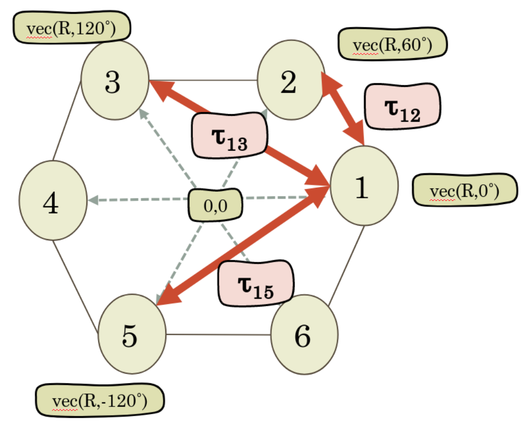

Fig. 1: In this example the center of formation is (0,0). Relative to the centre, τ_1 for agent 1 can be represented in polar coordinates as a vector with magnitude R and direction 0 °. Similarly, τ direction for agents 2, 3 & 5 can be encoded with 60°, 120°& -120°. τ_{rel} for each agent can be seen for $\tau_{12}, \tau_{13}, \tau_{15}$

Therefore, the formation specification can be formulated as $G_f = (V, E_f)$ for any arbitrary formation **f**, where E_f is encoded by τ_{rel}. The following assumptions have to be fulfilled when designing the formation configurations in order to leverage barrier functions introduced by Han et al. [1]:

1) The set of relative inter-agent distances τ must encode a feasible formation where the distances between the agents $d_{ij} = ||\tau_i - \tau_j|| > 0$
2) $E_f \subseteq E \;\forall\; T \geq 0$ i.e. the edge-set E_f for the formation specification is a subset of the edge set E for all time T.
3) When defining τ_{rel}, maintaining connectivity with the proximity radius r_s should not be at the cost of squeezing an agent anywhere into the safety-distance region of another one. Therefore,

$$r_s - ||\tau_{ij}|| > d_s + ||\tau_{ij}|| \;\forall\; (i,j) \in Q$$

For the sake of brevity, more information about the barrier function used in this work has been omitted.

Lastly, for the leader-follower system, the state information contained vertex v_i for an individual agent i $\in Q$ can be defined as follows:

1) Position vector: $X_i(t) \in R^p$, where p is the dimension of the state space.

2) Formational position vector: $Y_i(t) = X_i(t) - \tau_i$, where $\tau_i \in \tau^{\mathbf{f}}_{rel}$
3) Relative position vector: $X_{ij}(t) = X_i(t) - X_j(t)$ where $(i, j) \in Q, i \neq j$
4) Relative formational position vector: $Y_{ij}(t) = Y_i(t) - Y_j(t), (i, j) \in Q, i \neq j$

B. Cost Function for Global Planner

A* and RRT* are grid-based and sampling-based path-finding algorithms respectively, used to compute minimum cost paths as seen in Hart et al. [2] and Perez et al. [3]. Nodes are assigned an f-value on the basis of their cost-to-come function g. Using this principle, a node encodes the following:

1) Leader position $X_L : (x_l, y_l)$
2) Formation configuration $\mathbf{f} \in F$
3) Follower positions $X_F = \{(x^i_{\mathbf{f}}, y^i_{\mathbf{f}}) \forall i \in Q\}$
4) Cost-to-come g
5) Parent P

However, the neighbourhood search in A* or RRT* is simplified to 3-dimensions for a system with any number of agents i.e. the search is only done on a discrete grid (x,y) i.e. over R^2 and over $\mathbf{f} \in F$. The role of the position of the followers in the node is to influence the cost-to-come function using the formation state prediction function that uses barrier functions. With this definition of a node, the cost-to-come is reformulated to inculcate two metric costs that are weighted with λ_1 and λ_2 respectively:

1) Formational Error: As discussed in Section II-A, it is possible to calculate the relative formational position vector Y_{ij} between any two agents i and j, the intuition behind which, is that it denotes how far apart two agents are (X_{ij}) in comparison to the desired distance τ_{ij} between them. Therefore, as can be seen in Figure 2, when agent 2 moves away from its desired position, the formational error between agent 1 and 2, 1 and 3 & 1 and 5 can be individually calculated as $||Y_{ij}||$. Thus, the total formational error E_f of a formation f is:

$$E_f = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} ||Y_{ij}||_2}{2} \quad |_{(i,j) \in Q, i \neq j},$$

and the scale of that particular formation is the standard deviation of the elements of τ: $\beta = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\tau_i - \overline{\tau})^2}$ Lastly, taking into consideration all agents and scale of the formation, the average generalized formational error can be written as:

$$g_f = \frac{E_f}{N^2 \beta} \tag{1}$$

2) Distance to Obstacles: When planning for the leader of a multi-agent system, it is essential that the formation associated with it be feasible i.e., the follower agents occupy states in the configuration space that are free of obstacles. Let the configuration space be defined as C and the configuration space containing obstacles be C_{obs}. One of the main incentives of reconfiguring an existing formation is avoiding proximity to obstacles. With this motivation, each agents' minimum distances to obstacles in the environment are computed to get a set c_o:

$$c_o = \{\text{Distance Check}(C_{obs}, X_i) | \forall i \in Q\}. \tag{2}$$

However, the influence of c_o on g depends on a safety distance o_s. To model this, an exponential probability density function is considered.

$$g_o = \frac{o_s - \min\ (c_o)}{o_s} \exp\left(-\frac{o_s - \min\ (c_o)}{o_s} \overline{(c_o - c_o^P)}\right) \tag{3}$$

where c_o^P is the distance from obstacles computed by the parent node P, $\overline{c_o - c_o^P}$ is the mean of displacement from obstacles when moving from parent P to the current node and min c_o is the minimum of distance of the explored node from obstacles.

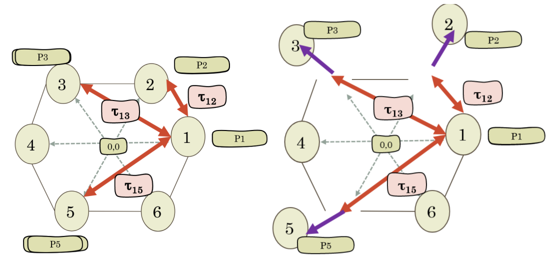

Fig. 2: **Left:** A hexagonal formation with τ_{rel} (red arrows) defined from agent 1 to 2, 3 and 5. State X of each agent is denoted with p. **Right:** After agent 2, 3 and 5, move in the direction of the purple arrows, their states X denoted by p changes. This demonstrates how formational error can be calculated.

C. Generating Neighbours

1) Clutter Bias: A clutter bias is introduced when exploring neighbours to build the tree in RRT*. When a random configuration of the system with $X_L$and$f$ is generated, it can be that X_L is collision free but corresponding X_F is not. These nodes are stored and re-explored with a different formation f to generate new X_F. The distance between agents in this new formation (e.g. radius of circle) has to be greater than d_s.

2) Follower States: Follower states are obtained by following the gradients returned by the barrier function due to the motion of the leader. Using the motion of the leader and that of the followers at every discrete step during the `Extend` subroutine of the RRT*, it is possible to check for both **feasibility** and **cost** of the path taken.

III. Preliminary Results

For the experimental setup, collision checking was done using the Flexible Collision Library, where the obstacles and robots were modelled as box objects with dimensions (3m, 3m, 2m) and (0.435 m, 0.456m, 0.32m) respectively. To validate the cost function, a narrow-passageway like environment is given where agents in a circular formation have to reconfigure. The specifications of the multi-agent system, cost function and

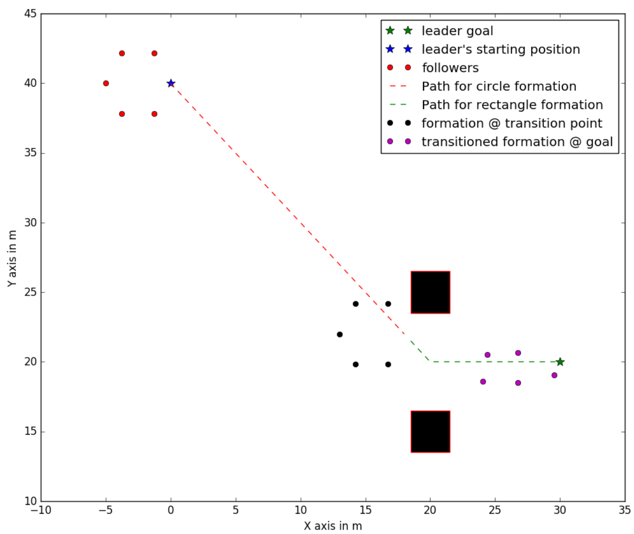

Fig. 3: Path planned for Environment 1

safety parameters were as follows: λ_1 = 0.88, λ_2 = 0.33, d_s = 0.5, d_c = 1, r_s = 20, r_c = 0.25, N = 6, o_s = 30m. The experiment returned the same path as A* did (for the leader), but the circle formation switches at about (18.5m, 20m) to a squeezed rectangle formation. The transition is possible and the green path in Figure 3 shows the path suitable only for the rectangle formation. Lastly, the goal is reached by the leader at (30m,20m) but the formation has changed to a rectangle one as seen by the followers plotted next to goal position.

IV. Conclusion

From experiments with both A* and RRT*, A* was inflexible in terms of the type of configurations explored and if the shape of a configuration is not in the search space, a solution cannot be found. However, the runtime of the neighbour generating step (with collision checking) was about 5 times that of A* for only the leader for upto 50 agents. The clutter bias introduced in RRT* helped ensure that solutions can be found by exploring over sizes of specified configurations. The cost function designed enabled a smooth transition by virtue of the distance cost only coming into effect when approaching obstacles and formational error preventing arbitrary fluctuations. Another advantage the usage of RRT* offers is in the parent-rewiring step of neighbours from the `Near` subroutine. When a new node (say P) is generated, we check paths from the nearest neighbour (say Q) to P and vice versa (P to Q). If P and Q have different shapes, the feasibility and costs of both these paths can be verified and will be different in case of proximity to obstacles.

References

[1] Dongkun Han, Lixing Huang, and Dimitra Panagou. Approximating the region of multi-task coordination via the optimal lyapunov-like barrier function. In *2018 Annual American Control Conference (ACC)*, pages 5070–5075. IEEE, 2018.

[2] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[3] Alejandro Perez, Sertac Karaman, Alexander Shkolnik, Emilio Frazzoli, Seth Teller, and Matthew R Walter. Asymptotically-optimal path planning for manipulation using incremental sampling-based algorithms. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4307–4313. IEEE, 2011.