Literature Review

Mason U'Ren

October 30, 2017

1 Rover Communication

Aaladdin: a Meta-Model for Analysis and Design of Organizations in Multi-Agent Systems [1]

The authors seak to create a interlinked rover communication network based around a two tiered obstraction model. They propose a *concrete level*, composed of the physical organization of the rover's core concepts (agents, groups, and role), and a second tier or *abstract/methodological level*, which defines all possible roles, valid interactions, and structure of groups and organizations. At the *concrete level*, agents are a member of a group, group contains a role, and role is handled by agents. While on the *abstract level*, group is contained by an organization which is instantiated from an organization structure, organization structure holds a group structure which can then instantiate to a group or define an interaction, interaction is defined between roles, and an agent is instantiated from an agent class. Tests were then run on an implemented "Mulit-Agent Development Kit" (MadKit) where they successfully validate their approach. By sticking to a basic philosophy, that the platform should handle its own management when working with agents, organized into groups, identified by roles, that the system supported distribution, multi-agents, and is scalable.

This concrete, yet complex communication and permission system architecture scales well to my research. Interlinked and "cognitive" rovers are the back-bone of multi-agent systems, thus by having a two tiered model built on simplicity and abstraction, we can confidently create any style of organization based around agents, groups, and roles. This gives us the freedom to tailor the model architecture to our needs.

Analyzing Myopic Approaches for Multi-Agent Communication [2]

In this paper, the authors establish an inter-rover communication threshold, which decides when to communicate information, when constrained by partial rover maps of their overall world. Communication actions are simply *communicate* or *don't communicate*, where if one agent chooses to communicate, all agents broadcast their local state to eachother at some cost. This synchronizes the world view generate by the rovers, providing complete information about the current world state. They develop a communication protocol, *Model Lookahead*, that only requires the rovers to hold the latest message of their projected world, instead of saving the entire message history, which ultimately saves times computationaly. Introducing and controlling messages in this way proved favorable as overcommunication of redundant information was decreased. In two different experimental setups, this approach produced smooth and monotonous degradation of values as communication costs increased.

This paper wraps up the general notion of computational cost that exists within multi-agent system. In order to implement a succinct, yet complex rover messaging system, like one presented in [1], inter-rover communications need to be held to a minimum. This will allow the system to be scaleable to excedeling larger size without computational slow down.

2 Agent Localization

Multi Agent Localization from Noisy Relative Pose Measurements [3]

Within the paper, the writers address the problem of estimating rovers' poses who do not share any common reference frames, when working with multi-agent systems. Their goal, in the presence of added noise, is to compute each agents pose, either relative to a specified anchor node or exclusively using neighbor-to-neighbor or local interations. The setup is to arrange independent rover interaction (poses) into a directed graph and have an anchor node placed at pose (0,0,0). Localization then is achieved in three phases: 1) an estimate of all the node orientations $\hat{\theta} \in \mathbb{R}^{n-1}$ in realation to the anchor, 2) then tangential position measurements of the nodes are expressed in terms of previuos iterations, 3) finally estimate pose positions and orientations are calculated using $\mathbf{p*} = ((\mathbf{x*})^T, (\theta^*)^T)^T$. Results proved favorable as both implementations converged to values within one standard deviation of the actual poses. An interesting aside found that the distributive approach converged under a wider range of classes of communications graphs.

Localization is extremely helpful when navigating in an unknown environment. This approach suggests accuracy of within centimeters, which means, in relation to my project, the rovers can have greater confidence in goal, resource, and holding positions. We can also wrap up rover-to-rover avoidance by relaying relative positions to one another.

3 Virtual Mapping

Distributed Localization and Mapping with a Robotic Swarm [4]

The writers discuss the issues native to location and virtual mapping within swarm robotics, when working with limited sensing and communication. Their work is motivated by designing a control algorithm capable of successfully completing a building-clearing mission, where the environment and object of interest are both unknown. Through exploration of the location, they hope to build virtual maps of traversed areas. The algorithm can be broken down into three parts which are colloborative localization, dynamic task allocation, and collaborative environmental mapping. The first labels the rovers *landmark* or *moving*, then uses infared sensors to measure and calculate therotical local poses, which are given different weights based on their classification and distance. Factoring individual odometry values into the equation, each pose is published into a goal frame with a particular confidence value. Dynamic task allocation handles the transition from *landmark* or *moving* laveled rovers. Since there is no external frame of reference outside of the system, the rovers take turns acting as *landmarks* while the others move collectively in the goal direction. *Landmarks* are seen as rovers with high confidence in poses, while *moving* rovers build confidence as they continual recalculate their position as they pass by *landmarks*. Task allocation occurs when enough confidence has been passed from a *landmark* to a *moving* rover at which point the *moving* becomes a *landmark*, and the *landmark* to *moving* if allowed by the system. Finally, the virtual map is constructed by forming triangles within the 2*d plane* consisting of two *moving* rovers as the base and the *landmark* as the point. If the rovers can see each other the are is marked seen by each rover involved, then pushed to a global map. Results were succesful in that rover locationization and mapping was achieve in an unknown environment, with or without obstacles, by augmenting this stepping stone approach.

Although not specific in the implementation, this stepping stone localization approach is interesting. But I believe establishing an anchor node before initial movement would guarantee higher pose confidence. Within this paper, their use of mapping would be the greatest contribution to my research. They have developed a simple, computationally, cost-effective way of building virtual maps on the fly which can be improved on to document resource and obstacles alike.

4 Foraging

Two Foraging Algorithms for Robot Swarms Using Only Local Communication [5]

The authors present two algorithms, when foraging for food, which allows coordination between robots, but don't require physical environmental marks such as pheramones. The two algorithms proposed, *Virtual Pheromones* and the *Cardinality Algorithm*, both arrange the robots into a dual class system of *walkers* and *beacons* and return two values based on relations to resource and the nest, but differ in the value assigned to each *beacon*. The first has the agents leave the nest, then begins to classify *beacons* adhoc. Floating point values are assigned to each beacon, with a constant decay rate, where rewards are given to *beacons* that are within range of *walkers*, thus creating a trail to the robot with the highest *virtual pheromone* value. *Cardinality* differs in that it doesn't calculate pheromone trails but counts the number of beacons within range between itself and the resource or the nest. These approaches were compared against a random walking algorithm and a food GPS algorithm that knows the location of resources and the nest alike, which creates a lower and upper bound for their experiments. They found that the *cardinality algorithm* outperformed the *virtual pheromone* in both obstacle and obstacle-free cases with a much higher food return rate and quicker model stabalization over fewer time steps. These algorithms were also scalable to an extent, but when the number of rovers exceeded 100, congestion was created around resources and the nest.

This article presents an interesting approach to foraging for materials, using the swarm as a whole to search larger areas. Although not directly specific to project, the rovers will need some type of foraging control algorithm that governs their trajectories. I can use this style of searching and path plotting between resource and nest within my project. Not explicitly discussed in the summary, but an interesting aside was their conclusion that two separate paths that share only goal and resource endpoints, was needed to eleviate head-on rover collusions when transporting food. I had a similar idea that had the rovers travel along directional arc that were equivalent, but rotated by $\frac{\pi}{2}$, that would lay between resource and home. This is similar to how a highway is setup.

References

[1] Jacques Ferber and Olivier Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Multi Agent Systems, 1998. Proceedings. International Conference on*, pages 128–135. IEEE, 1998.

[2] Raphen Becker, Alan Carlin, Victor Lesser, and Shlomo Zilberstein. Analyzing myopic approaches for multi-agent communication. *Computational Intelligence*, 25(1):31–50, 2009.

[3] Rosario Aragues, Luca Carlone, G Calafiore, and C Sagues. Multi-agent localization from noisy relative pose measurements. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 364–369. IEEE, 2011.

[4] Joseph A Rothermich, M İhsan Ecemiş, and Paolo Gaudiano. Distributed localization and mapping with a robotic swarm. In *International Workshop on Swarm Robotics*, pages 58–69. Springer, 2004.

[5] Nicholas R Hoff, Amelia Sagoff, Robert J Wood, and Radhika Nagpal. Two foraging algorithms for robot swarms using only local communication. In *Robotics and Biomimetics (ROBIO), 2010 IEEE International Conference on*, pages 123–130. IEEE, 2010.