Summary

In the past two weeks I have coded up a simple simulation in which robot agents move around in the map looking for the best site out of 2. This serves as a simple baseline model in which there is no faulty robots and no evolution to improve the performance of the swarm. We could then compare this result with the case when there are faulty robots in the population and as we kick start evolution.

Recall our reward was defined as

$$reward = \frac{\text{no. of robots correct}}{\text{no. of robots in swarm}}$$

The following figure shows the average reward over time in robot swarms that have no adversaries.

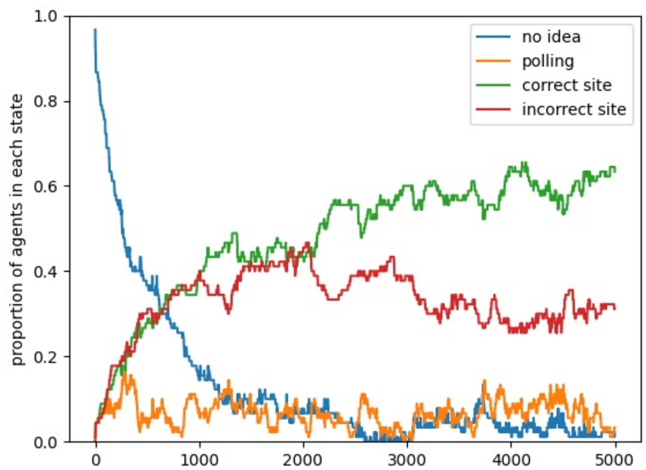

Figure 1: Average reward over time when there is no faulty robot in the swarm, across 10 independent runs

On top of this, I introduced the idea of "faulty" robots into the model. Specifically, some number of the robots will give a false quality information to others during communication. We are interested in finding out the effect of this "lying" behavior to the performance of the robot swarm. In expectation, this should lead to confusion of robots, however, we are wondering if it could encourage more exploration and adaptation when evolution is introduced.

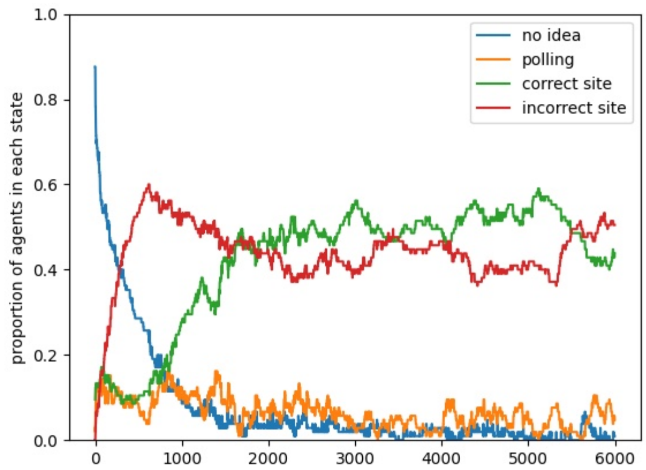

Figure 2: Average reward over time when there is 1 faulty robot in the swarm

As shown in figure 2. with only 1 faulty adversary, we observe the case in which the majority belief switch from the correct site to the incorrect side, despite the majority having the correct belief earlier. This suggests that our previous hypothesis of faulty robot leads to confusion is probably correct, but needs to

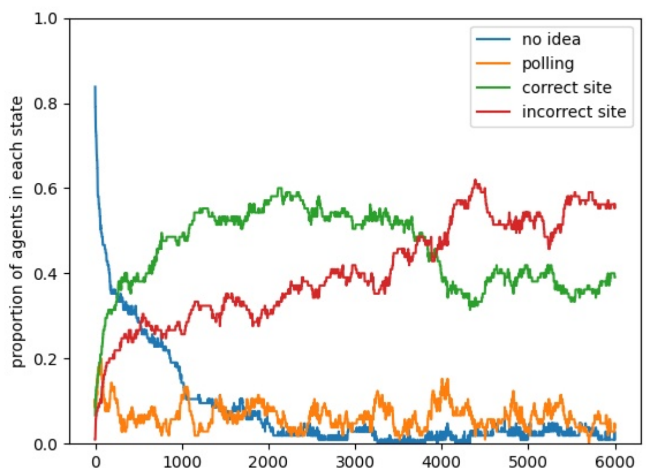

Figure 3: Average reward over time when there is 5 faulty robot in the swarm

When there are more faulty robots in the population, the effect of confusion is more obvious. We can see here that although the correct belief have a clear win until time step about 3500, the incorrect belief managed to win later on.

To-Dos

1. As can be seen from all 3 graphs, the initial course of development seems very random. (Currently, the behavior from 0 to 2000 timesteps are extremely random and could affect the course of development later on). There is also the case of *drift* happening, during which the majority have the incorrect belief, making it hard for the correct belief to propagate.
   We are still in the process of trying to find a good initial condition to minimize noise and other factors. I have tried some, such as fixed starting point, but they seem to lead to worse result.

2. Although we are averaging over a couple of runs, the curve is never smooth, it is thus part of the plan to increase the number of simulation runs I am currently running to produce more convincing result.

3. I am currently in the process of writing code for the evolutionary process. Hopefully that could lead to something interesting.