Exploration of Collective Perception in presence of Lying Agents

Shubham Jain, Yasmina Benkhoui, Sanket Gujar

Abstract— **Collective perception for best-of-N problems in swarm robotics involves choosing the best option out of N alternatives in a decentralized manner. We aim to investigate the effects of lying and faulty individuals on collective perception using Bayesian inference. We will focus on recognizing a floor pattern made up of black and white, using a swarm of robots. Experimentation will be done on the number of lying individuals, decision making strategies of the agents and the inter-relation between them**

I. Introduction and Background

Collective decision making or collective perception is a very important application of swarm robotics where groups of robots collectively reach a decision based on their individual beliefs. However, once this decision has been made, it cannot be traced back to a single individual agent from the swarm. This behaviour can be found in nature especially in bees and ants where, for example, they collectively decide the place for building their nest. Alternatively, in the context of social insect colonies, collective consensus is used to allocate different tasks to subgroups of agents as per the requirements of the colony.

Researchers have tried to replicate such behaviours using swarm robots [5]. They can have many applications like search and rescue, exploration and so on when the area to be explored is very large. In our project, the main focus would be to study collective decision making for achieving consensus. We aim to investigate the robustness of such collective perception techniques even in the presence of faulty or lying agents.

Faulty agents refers to those agents in the swarm that are malfunctioning due to issues such as insufficient current flow in the robot, hardware faults, inaccurate sensors and so on. On the other hand, lying agents refer to malicious agents that are deliberately giving faulty values to attack the network. [3] explores how individuals develop a tendency to lie in a competitive scenario for their own gain. This work uses genetic algorithms to show that robots in a swarm might lie in competitive situations to confuse other agents and gain advantage. Hence the goal of our project is to explore how collective perception algorithms are affected by the presence of such deceptive agents.

One possible way of detecting faulty agents could be to build fault tolerance into the swarm. This is done by enabling them to identify flawed individuals from their neighbours to compensate for them. For instance, [2] presents a method inspired by synchronised flashing behaviour of fireflies where all robots flash synchronously after some time unless they are malfunctioning. Asynchronised flashing of the faulty individuals allows their neighbours to identify them. Such a method is called exogenous fault detection. However, for our project, we will explore algorithms which inherently deal with faulty individuals rather than explicitly trying to identify them.

We will be considering the decentralised best-of-n problem for cooperative decision making as posed in [4]. Here, robots must try to collectively choose an alternative out of N different solutions available to them. One approach to this problem is to consider a third truth state as proposed by [1] where the robot has an option of either choosing one of the alternatives or being in an unsure state. This makes collective decisions more robust to lying agents who are trying to confuse the network.

II. Proposed Work

Can a swarm of robots recognize floor pattern given a set of templates by sensing and communications in a decentralized manner.

We propose to experiment with swarm of khepera robots which have sensing to detect the color beneath it and able to exchange messages with its neighbors. The templates will have binary state for each position and the individuals given its position and neighbors prediction will have to broadcast the template pattern which it believes is present in the environment. The challenge comes when the swarm is introduced with a lying individual which will broadcast false signals to its neighbors. The swarm will have to take the correct decision in presence of noise.

The following setup will be used to carry out the experiments:

1) The robot will know its position in the pattern at all time.
2) Robots can move in the following manner
   a) Random motion.
   b) Non-random motion (Following a certain pattern/ dictated by external process).
3) Noise can be added to the sensing and actuators.

The robots can use average consensus on each pattern template sensed by the robot. The robot can finally estimate the one having the largest probability.

We can also use weighted sum of estimates, where the unsure robots have low weights and robots that are sure can have a higher weights.

III. Proposed Experiments and Expected Outcomes

The experiment can be carried out in the following manner:

- The Lying individual can broadcast the following faulty messages to its neighbors.
    1) Wrong estimate (sensor fault).
- The number of lying robots can be increased to study the resultant behavior till the system breaks.
- The number of available templates to the robots can be varied to study the distribution of the decisions.
- The complexity of the templates can be increased (Asymmetric patterns).
- Noise when added to actuation instead of the sensing.

IV. Algorithm

Symbols

y^l	l th pattern
x_t	Robot position at time t
z_t	Robot's observation at time t
$neigh$	no. of neighbors
$neighbelief_t^i$	i_{th} neighbours belief message at time t
$belief_t$	belief distribution of the robot at time t

Algorithm 1 Decentralized Pattern Detection

for `no of iterations` **do**
  for `every robot` **do**
    $neighbelief_t \leftarrow$ **receive()**
    for `all patterns` **do**
      $\overline{belief_t^l} \leftarrow (\prod_i^k neigbelief_t^{l,i})belief_{t-1}^l$
      $belief_t^l \leftarrow p(z_t|y^l, x_t)\overline{belief_t^l}$
    $belief_t \leftarrow$ **normalize**$(belief_t^l)$
    broadcast$(belief_t)$

Algorithm 2 Decentralized Pattern Detection(Avg. Belief)

for `no of iterations` **do**
  for `every robot` **do**
    $neighbelief_t \leftarrow$ **receive()**
    for `all patterns` **do**
      $\underline{neighavg} \leftarrow ((\sum_i^k neigbelief_t^{l,i})/neigh)$
      $\overline{belief_t^l} \leftarrow (neighavg)belief_{t-1}^l$
      $belief_t^l \leftarrow p(z_t|y^l, x_t)\overline{belief_t^l}$
    $belief_t \leftarrow$ **normalize**$(belief_t^l)$
    broadcast$(belief_t)$

V. Setup

The simulation has been setup in ARGoS where we consider a 4×4 grid of black and white tiles and 20 uniformly distributed khepera robots as seen in figure 1. The task for the robots is to find the right pattern out of different templates that are given to them The robots have three sensors -

Algorithm 3 Decentralized Pattern Detection (SSR) [6]

for `no of iterations` **do**
  for `every robot` **do**
    $neighbelief_t \leftarrow$ **receive()**
    $weightedbelief_t \leftarrow sim(belief_t, neighbelief_t)$
    $weightedbelief_t \leftarrow sort(weightedbelief_t)$
    $neigbelief_t^i \leftarrow$ remove n neighbours having lowest weighted belief
    for `all patterns` **do**
      $\overline{belief_t^l} \leftarrow (\prod_i^k neigbelief_t^{l,i})belief_{t-1}^l$
      $belief_t^l \leftarrow p(z_t|y^l, x_t)\overline{belief_t^l}$
    $belief_t \leftarrow$ **normalize**$(belief_t^l)$
    broadcast$(belief_t)$

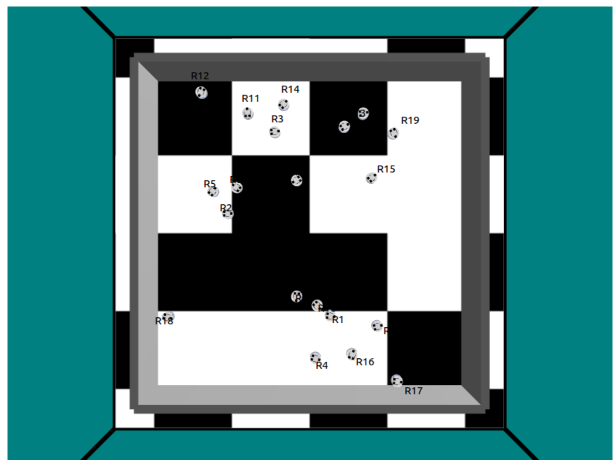

Fig. 1. Experimental setup for a 4x4 grid with 20 robots in ARGoS simulator

We have implemented the algorithm mentioned above in Python for 4 robots with 3 templates with some simplifications like stationary robots and full connectivity for communication. We are now working on implementing the same using Buzz for moving robots with 20 moving robots, communication only with local neighbours and higher number of templates.

VI. Experiments

The initial experiments are carried out in two settings:

- Robots without noise.
- All the robots have sensor noise.

For the experiment, We started with 4 robots on template of size 4. The robots were given 3 templates to choose from. They have a equal starting probability for every pattern, and the robot is communicating with every other robot present for 10 timesteps before making the final decision.

A. Without noise

Here the robot has no sensor noise and gives the correct template color. Here the sensing update was made by assuming:

Ground sensor	Returns a value of 1 when the robot is on a white tile and 0 for black tile
Position sensor	Gives the position of the robot in the world frame
Proximity sensor	Used for obstacle avoidance

$$p(z_t|y^l, x_t) = \begin{cases} 1 & \text{if} \quad y^l(x_t) = z_t \\ 0 & \text{otherwise} \end{cases}$$

without noise the robots converges to the correct decision with maximum probability every single time.

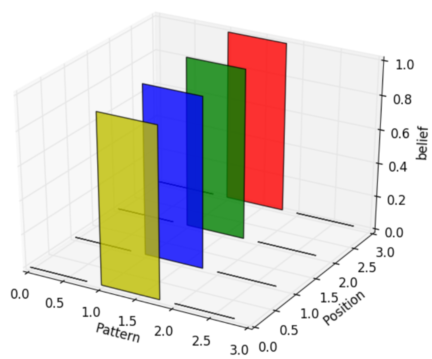

Fig. 2. **Without Noise**: Probabilistic decision distribution of robots for correct template no. 2 after 10 timesteps

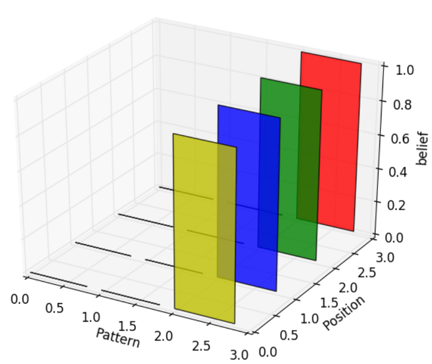

Fig. 3. **Without Noise**: Probabilistic decision distribution of robots for correct template no. 3 after 10 timesteps

B. With Noise

The settings here are the same as the previous experiment but here we considered the sensor has a probability to give us the wrong reading with probability w. Also, all the robots sensor are noisy.

$$p(z_t|y^l, x_t) = \begin{cases} 1 - w & \text{if} \quad y^l(x_t) = z_t \\ w & \text{otherwise} \end{cases}$$

we tried varying he noise probability w to check the effect on the decision of the robots. We got the result that if the sensor has a noise probability less than 0.5 then it will converge to the right decision every single time.

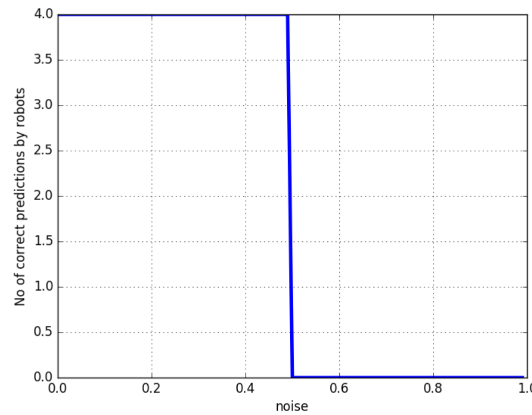

Fig. 4. **With Noise**: Effect of sensor noise on decision of the robots

C. Lying Individuals

We propose experimenting with adding lying individuals into the network which can be of the following types -

- Type 1 - Robots update their beliefs according to the opposite of the colour that they observe
- Type 2 - Robots update their beliefs according to one specific colour all the time irrespective of the reading given by its ground sensor

We propose studying the effects of the overall belief of the system as the number of lying individuals is increased and with the different type of lying individuals.

REFERENCES

[1] Crosscombe, M., Lawry, J., Hauert, S., Homer, M. (2017). Robust distributed decision-making in robot swarms: Exploiting a third truth state. In IEEE/RSJ international conference on intelligent robots and systems (pp. 43264332). Vancouver, British Columbia, Canada: IEEE.

[2] A. L. Christensen, R. O'Grady, M. Dorigo, "From Fireflies to Fault-Tolerant Swarms of Robots", IEEE Trans. Evol. Comput., vol. 13, no. 4, pp. 754-766, Aug. 2009.

[3] https://spectrum.ieee.org/automaton/robotics/artificial-intelligence/swarm-robots-evolve-deception, EPFL.

[4] A C Parker, Chris and Zhang, Hong. (2009). Cooperative Decision-Making in Decentralized Multiple-Robot Systems: The Best-of-N Problem. Mechatronics, IEEE/ASME Transactions on. 14. 240 - 251. 10.1109/TMECH.2009.2014370.

[5] T. Schmickl, C. Mslinger, K. Crailsheim, Collective perception in a robot swarm, in: Swarm Robotics, Springer, Berlin, 2007, pp. 144157.

[6] K. Saulnier, D. Saldana, A. Prorok, G. J. Pappas and V. Kumar, Resilient Flocking for Mobile Robot Teams, IEEE Robotics and Automation Letters, Vol. 2, No. 2, pp. 1039-1046, April 2017.