State of the art

Eric Mas Moncusi

Article 1

Cooperative leader following in a distributed multi-robot system

In this paper they adress the probem of managing a group robots with hetereogeneous capabilities. The first robot navigates by using a predefined pattern or by being teleoperated, whie the other robots follow it. To achieve that, a series of rules are needed, at team and individual level (both levels can be executed in parallel). Those behaviours are:

- **Team - Follow:** every robot follows the local leader.
- **Team - Wait:** If an unexpected event happens the group stops.
- **Team - Recover:** It is executed to recover from a wait condition.
- **Robot - Follow:** The robot its following the leader.
- **Robot - Local wait:** The robot waits due to an obstacle.
- **Robot - Remote wait:** The robot is waiting because other one or more robots are in local wait.
- **Robot - Local recover:** The robot is trying to overtake an obstacle and follow its leader.
- **Robot - Remote recover:** The robot is still following the leader, but at a reduced speed, so that if another robot is doing a robot llocal recover it will be easier for it to perform its task

At a team level behaviours are only triggered by communication, whereas local ones can be both executed by sensors or communication. The agorithm would perform the same, independently of the size of the team. State diagram for team and robot behaviours here:

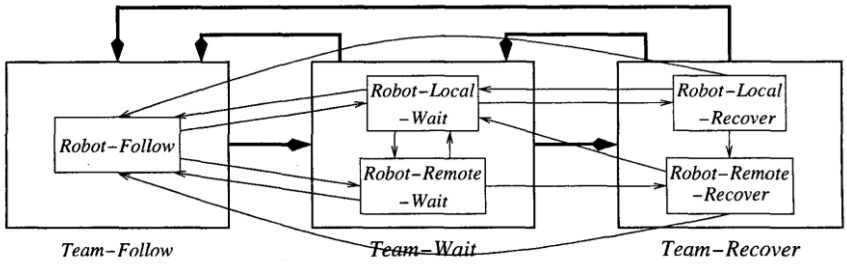

Figure 1: State diagram for team and local behaviours

To enabe communication between the team global counters are used for wait and recovery states. Everytime a robot faces a dangerous situation both counters are increased and a timer starts, when the timer ends the wait counter decreases. Finally, when the danger is over the recovery counter is decreased too. These counters provide additional feedback to the other robots and enhance their sensorial information. If at some point a robot would fail in the middle of a dangerous situation the other robots have a timer that enabes them to sense that and resume operations after a certain time. The software was executed with a group of 3 to 5 mobile patforms. Each one equiped with different sensors and with different characteristics. Moreover, it was buit as a muti-threaded program, so each sensor has its own threat and then the output is sent to another threat which decides based on a set of fuzzy rules. The communication aso has its specific threat. Finay, to recognise and track the eader a combination of CCD image sensor and aser scanner is used. The laser enabes the robot to sense the minimum distance and the CCD alows to detect the shape of the robot to follow with the following sequence:

1.- **color segmentation:** searches a charachteristic coor of the robots.

2.- **Averaging (smoothing):** Depends on neightbours, if four or more are red, then red. Otherwise white.

3.- **Blob detection:** Examines regions by boundaries of red pixels.

4.- **Object assignment:** Diferent label for each connected component.

5.- **Object selection:** Decides which of the objects should be track by comparing the center of mass of every distinct object with the positon of the previous tracked robot.

6.- **Proximity estimation:** Depending on the size of the blob it estimates the distance to the leader