ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

WS 2014-2015, STI-SMT, No. DISAL-SP68 Project Summary

Dealing with Failures in Real Quadrotor Formations

Rémy Siegfried

Professor : Alcherio Martinoli
Assistant(s) : Duarte Dias

This semester project was about making experiments on real quadrotors formation safer. The first goal was to improve the way to command multiple robots at the same time and to enable the formation control from an external computer. As failures in the system can lead to unpredicted and potentially dangerous behaviours of the robots, the second goal was to make the quadrotors detect the failure and react to it in order to avoid damaging the robots and their environment.

The first part was achieved by letting the user create teams with robots, which can be commanded with one single command. Then, all the information related to the formation that should be applied was stored in a text file that the robots can interpret. In this manner, the user can easily send a new text file to the robots and change the formation.

In order to structure the implementation, the second part of the project was approached by listing all the failures that can occur in a real quadrotor formation. It leads to a huge list of case that was helpful to figure what can happen but didn't give a structure for the code. This approach was let apart and a new point of view was investigated by considering the local effect of the failures on each robot separately. It leads to smaller list of events, resumed by six major cases; a robot can: lose its position, lose the position of another robot, lose the position of the leader, runs out of power, detect that another robot behave weirdly or detect that a collision with another robot is near.

From here, each case was related with a safety policy that the robots must respect. For example: if a robot runs out of power, it musts land. It leads to a rule based behaviour structure that can be implemented on the quadrotors. As each rule has different consequences and use different information in order to detect failure, each one can be implemented separately of the others, giving a very flexible architecture. The idea is that each robot can trigger a different rule corresponding to its local perception in order to fix the global situation. For example: a robot runs out of power and land, when another robot detect that it behaviour weirdly, because it goes down, and cut all connections with it. In the end, the failing quadrotor is at the ground and the remaining robots ignore it completely and continue to fly.

The rules concerning the loss of position, the power supply and the detection of weird behaviour were implemented and tested in several experiments, showing good qualitative results. The quantitative results were difficult to analyse, because a quadrotor encountered problems to fly, which can't be fixed before the end of the project.

In conclusion, the command of the system was successfully improved. In the field of safety policies, the architecture explored in this project seems appropriate to multiple agent systems, because it is highly flexible and take in account the local situation of each member, even if the concrete way to detect, as well as the chosen reaction, could be investigate further in order to be more robust to less artificial environment.