ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1720

Using Petri Net Plans for Modeling UAV-UGV Cooperative Landing

Andrea Bertolaso and **Masoume M. Raeissi** and **Alessandro Farinelli** and **Riccardo Muradore** [1]

Abstract. Use of cooperative multi vehicle team including aerial and ground vehicles has been growing rapidly over the last years, ranging from search and rescue to logistics. In this paper, we consider a cooperative landing task problem, where an unmanned aerial vehicle (UAV) must land on an unmanned ground vehicle (UGV) while such ground vehicle is moving in the environment to execute its own mission. To solve this challenging problem we consider the Petri Net Plans (PNPs) framework, an advanced planning specification framework, to effectively use different controllers in different conditions and to monitor the evolution of the system during mission execution so that the best controller is always used even in face of unexpected situations. Empirical simulation results show that our system can properly monitor the joint mission carried out by the UAV/UGV team, hence confirming that the use of a formal planning language significantly helps in the design of such complex scenarios.

1 Introduction

There are a wide variety of applications that take advantage of cooperative multi vehicle team including aerial and ground vehicles. Search and rescue[3], target detection and tracking[6] and mines detection and disposal [1] are a few examples of such applications that benefit from collective behavior of different types of unmanned robots.

In this work we consider a cooperative control scenario, where the UAV/UGV team should operate in tight cooperation to perform a joint task. In particular, here we focus on a cooperative landing scenario, where an unmanned aerial vehicle (UAV) must land on an unmanned ground vehicle (UGV) while such ground vehicle is moving in the environment to execute its own mission. Our goal is for the UAV to perform a fast and safe landing maneuver, hence we propose a strategy where the UAV quickly approaches the UGV and then carefully plans a safe landing trajectory.

A crucial open issue for multi robot systems that perform tight cooperation is to recover from possible failures due to unexpected events. For example, consider a situation where the UAV is initiating the landing maneuver based on the future positions communicated by the UGV. If the UGV must suddenly change its current trajectory (e.g., due to a moving obstacle) the UAV should smoothly adapt its plan to recover from a possible failure.

In this paper we investigate the use of high level languages or team plans [5, 4, 7, 2] to describe and monitor the activities of vehicles during mission execution to achieve the collective behaviors and goals even in face of such unexpected events. Specifically, our focus is on Petri Net Plans (PNPs) framework [7] to specify the collaborative landing task. There are several benefits related to the use of the PNP framework: first it provides a rich graphical representation that helps the designers to create plans with minimal effort, second the generated plans can be monitored during the execution, third PNPs support well-defined structures for handling tight coordination and on-line synchronization in multi robot systems.

Finally We evaluate our approach in V-REP, a realistic simulation environment, using state of the art tools for robots control. Our experiments show that the proposed approach can effectively monitor the cooperative behavior of the two vehicles recovering from possible failures.

2 UAV/UGV Cooperative Landing Scenario

The problem addressed in this paper is a particular kind of collaboration between heterogeneous autonomous vehicles: the landing of an UAV on an UGV. The collaboration task is composed of three phases:

1. both the UGV and UAV are moving according to their specific and non-cooperative tasks;
2. the UAV approaches the UGV (*flyFar* action using the PNP terminology);
3. the UAV lands on the UGV (*flyClose* action using the PNP terminology).

In Phase 2 the UAV is using its sensing system (e.g. camera) to locate the UGV and plans the faster trajectory to approach the UGV. In this phase the UGV in not aware of the intention of the UAV and so it is continuing its task as in Phase 1. In Phase 3, the UAV is close to the UGV and information are exchanged between them: the UGV is getting aware of the intention of the UAV and so it decreases its velocity and sends to the UAV its planned trajectory to easier the landing. This means that the UGV is still pursuing its objective (e.g. patrolling an area) but in a slower way.

We used JARP to create the Petri Net plan, however any of the available graphical tool that supports pnml (Petri Net Markup Language) file format could be used. The simplified version of the plan is shown in figure 1(b).

Actions name and all external conditions have been defined in the plan. Actions represent robot behaviors, for example in our case the *flyFar* action represents the UAV flying towards the UGV constantly following its position. In addition to the actions, we use interrupt operator, an important structure of PNP, to model action failures and activate recovery procedures based on the occurred condition. Conditions are external events and need to be checked at run time. The possibility to define conditions is a powerful feature that allows to enrich the plan behavior at run time.

[1] Computer Science Department, University of Verona, Italy, email: andrea.bertolaso@studenti.univr.it, masoume.raeissi@univr.it, alessandro.farinelli@univr.it, riccardo.muradore@univr.it

The plan execution will be started by initializing the UGV and then the UAV. Both robots will execute the *moveFar* and *flyFar* actions, until UAV gets close to the UGV or decides to landing. The close and far distances are application-dependent. When the UAV is close to the UGV, the *flyFar* action is interrupted and UAV sends the *close* event to the UGV. The UGV *moveFar* action is interrupted as well.

When the vehicles are getting close, UAV's behavior should be changed based on UGV's future position. UAV must be informed of UGV's future position to coordinate their actions. Thus UAV will receive UGV's future position periodically (T seconds). Every time a new future position is sent to the UAV, this new position will interrupt the *flyClose* action so the UAV can recompute flight trajectory in order to follow the new location of UGV.

After the plan is designed, we have to hand coding actions and conditions by using ROS [2] actionlib interface and makes them available to the PNPros which connects the PNP executor with ROS. Then PNP executor processes the Petri Net (pnml file described above) and executes the plan within the ROS.

3 Simulation and Evaluation

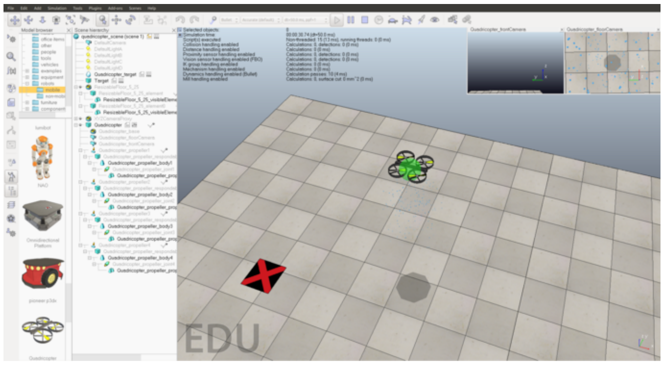
For running the experiments, we create in V-REP a simulation environment containing the UAV and the UGV. The initial position of both vehicles can be chosen arbitrarily in order to obtain different experimental setups. Figure 1(a) shows an initial positions of UAV and UGV in V-REP environment. Communication with V-REP is possible through ROS topics. When the simulation environment and the system that handles the plan are launched, the initial position of UAV and UGV is retrieved from V-REP via ROS topics.

The actual positions of the UAV and UGV are communicated to V-REP during the execution of the plan. The simulation environment will be updated according to the new changes. The whole system keeps on running until a final state in the Petri Net plan is reached. Figure 1(b) shows a snapshot of the simulation when the UAV is flying toward the UGV (*flyFar* action during Phase 2).

A video showing the complete execution of the plan can be seen at the link in the footnote[3]. The video illustrates that the coordination between the two vehicles is not a one-step synchronization action but it is a continuous behavior. The vehicles start far away from each other; then the UAV flies toward the UGV with maximum speed (Phase 2) until the close condition comes true (Phase 3). At this moment, UAV sends an external event to the UGV and the UGV starts sending its next position to the UAV. UGV decreases its speed to make the landing easier. The future position of the UGV is important for the UAV because unexpected events may happens (e.g. obstacles) that prevent the UAV to land on the UGV and so they may get far away again (from Phase 3 to Phase 2). The video also shows the evolution of the simplified version of the Petri Net plan during the simulation in order to better illustrate the behavior of the system. The mission is accomplished when the UAV lands on the UGV: this corresponds to the final state (place) of the plan.

ACKNOWLEDGEMENTS

This work was supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 689341. This work reflects only the authors' view and the EASME is not responsible for any use that may be made of the information it contains.



(a) Initializing UAV and UGV in the environment

(b) The Petri Net plan demonstrates the situation when UAV flies toward UGV

Figure 1. V-REP environment setup for simulating the cooperative landing task.

REFERENCES

[1] L. Cantelli, M. Mangiameli, C. D. Melita, and G. Muscato, 'Uav/ugv cooperation for surveying operations in humanitarian demining', in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–6, (Oct 2013).

[2] Alessandro Farinelli, Nicoló Marchi, Masoume M. Raeissi, Nathan Brooks, and Paul Scerri, 'A mechanism for smoothly handling human interrupts in team oriented plans', in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pp. 377–385, (2015).

[3] M. K. Habib, Y. Baudoin, and F. Nagata, 'Robotics for rescue and risky intervention', in *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*, pp. 3305–3310, (Nov 2011).

[4] Gal A. Kaminka and Inna Frenkel, 'Flexible teamwork in behavior-based robots', in *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, pp. 108–113, (2005).

[5] Milind Tambe, 'Towards flexible teamwork', *Journal of Artificial Intelligence Research*, 83–124, (1997).

[6] H. Yu, R. W. Beard, M. Argyle, and C. Chamberlain, 'Probabilistic path planning for cooperative target tracking using aerial and ground vehicles', in *Proceedings of the 2011 American Control Conference*, pp. 4673–4678, (June 2011).

[7] V.A. Ziparo, L. Iocchi, PedroU. Lima, D. Nardi, and P.F. Palamara, 'Petri net plans', *Autonomous Agents and Multi-Agent Systems*, **23**(3), 344–383, (2011).

[2] http://www.ros.org/

[3] https://goo.gl/hiZIKP