Flexible POMDP Framework for Human-Robot Cooperation in Escort Tasks

(Extended Abstract)

Fabio-Valerio Ferrari[*]
GREYC, University of Caen Normandy
Caen, France

Laurent Jeanpierre[†]
GREYC, University of Caen Normandy
Caen, France

Abdel-Illah Mouaddib[‡]
GREYC, University of Caen Normandy
Caen, France

ABSTRACT

We describe a novel method for ensuring cooperation between human and robot. First, we present a flexible and hierarchical framework based on POMDPs. Second, we introduce a set of cooperative states within the state-space of the POMDP. Third, for ensuring an efficient scalability, the framework partitions the overall task into independent planning modules. Lastly, for a robust execution of the POMDP policies we use Petri Net Plans, which have already been used to execute MDP policies. To this end, we describe how to convert a POMDP policy into an executable Petri Net Plan. We implement our approach and develop experiments on simulation and on a real robot in an escorting task where the robot guides a customer to the desired place in a public space.

1. INTRODUCTION

Autonomous robots have to face several issues in order to perform their task in crowded public spaces such as malls[6], hospitals and museums[1]. Service robots often need to interact and cooperate with humans in a dynamic and unpredictable environment, for example to initiate and achieve a task. This kind of cooperation, however, has a weaker level of commitment w.r.t. other application fields, such as cooperation with professional workers in an industrial environment. We aim at generating policies that account for the uncertainty of the human's behavior and his possible lack of commitment to a shared task directly within the planning phase and at providing methods for ensuring a robust execution of such policies.

2. PROPOSED FRAMEWORK

The proposed model has a hierarchical structure based on three independent layers, following the framework proposed in [2]: a Primitive layer, a Cooperation layer, modeled as a Partially Observable Markov Decision Process (POMDP)[8] [5], and a Task Status layer. The hierarchical structure is built through an abstraction process, from the Primitive layer (less abstracted) to the Task Status layer (more abstracted) (Figure 1). Within the Cooperation layer, we define two sub-systems, each meant to perform planning on a subset of the Cooperation layer state-space.

2.1 Primitive layer

The Primitive level takes information from sensors and executes low-level reacting. Actions taken at this level are *primitive actions*. During execution, it provides observations and decomposes macro-actions provided by the upper layer into primitive actions. In the Escort Task application it mainly performs robot navigation, but it also needs to generate spoken dialogue plans when interacting with users.

2.2 Cooperation layer

The Cooperation level abstracts the state space of the Primitive layer in order to focus on the variables which define the current mental state of the human. It generates plans of *macro-actions* for both achieving the task goal and re-establishing cooperation when needed.

We adopt a model similar to [7] and divide our Cooperation layer into two sub-systems: the *Cooperation* and the *Task* systems. While the latter aims at achieving the mission goals, the first aims at re-establishing the human-robot cooperation and reinforce the beliefs over his intentions. The decision as to which sub-system activate and hence which action perform during execution is taken at the Task Status level. Given the set of state variables at the Cooperation layer, we define two groups:

Task variables pertain to the overall task, regardless of the human's level of commitment and mental state. These variables are *irrelevant* for establishing the status of cooperation among agents.

Cooperation variables are those variables which have been introduced explicitly to deal with the human's commitment. For the Escort Task application, we use the following features as Cooperation variables:

Attention level: We define three values to determine how much the human is concentrated on following the robot: *Focused, Distracted* and *Lost*. They can be estimated by performing face tracking on the cameras data and checking whether the human is looking at the robot or not.

[*]fabio-valerio.ferrari@unicaen.fr
[†]laurent.jeanpierre@unicaen.fr
[‡]abdel-illah.mouaddib@unicaen.fr

Appears in: *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

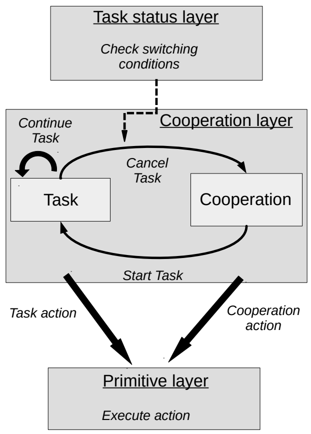

Figure 1: Structure of the framework

Proxemic Interaction Distance: From the camera data, we compute the human's coordinates and use the human-robot distance to estimate his state of cooperation. We use the study on Proxemics [3] to define a set of human-robot social distances: *Intimate, Personal, Social, Public.*
Human Relative Position: Still from the human's coordinates, we use a discrete set of positions of the human w.r.t. the robot, such as Left, Right, Front etc.

For more flexibility, we model the Cooperation module as a POMDP independently from the Task module.

2.3 Task status layer

The Task status layer checks which is the current overall state of the system and chooses whether executing actions from the Task or Cooperation sub-system. At it simplest, it consists of a set of execution rules and *switching conditions* allowing to switch from one sub-system to the other. At each execution step, the system updates its belief on the state of Cooperation variables. Within the state-space S, we define a set of Cooperative states $CS \subset S$ where the joint intention between human and robot is preserved. As long as the current state $s \in CS$, then all agents are trying to achieve the common goal: everything is going well and the robot can proceed with the task. Otherwise, the agent needs to bring the system back to a CS state. In other words, CS is the set of states where no action is required specifically to repair the missing cooperation between human and robot. Whether s belongs to CS or not depends on the current values of Cooperation variables. If $s \in CS$, then the system activates the Task sub-system which will provide the action to be executed. Otherwise, the system will select the action from the Cooperation sub-system's policy to ensure cooperation.

3. IMPLEMENTATION AND RESULTS

3.1 Implementation

We implemented the presented framework for the Escort Task scenario. The Cooperation module POMDP was implemented as a discretized Belief-MDP and solved using Value Iteration. In order to ensure robust execution of the generated policy, we implemented the Primitive layer as a Petri Net Plan (PNP)[9]. We have extended the approach presented in [4], which describes how to translate a MDP policy into a PNP, to POMDP policies. By solving the Belief-MDP, we obtain a policy π which associates an action a to each belief point b of the discretized Belief-MDP. Then we explicit the outcoming belief point b' for each observation o, with action a given by the Belief-MDP policy. We can therefore implement the policy as a PNP, by treating the POMDP observations as conditions for PNP transitions and the Belief-MDP belief points as PNP states.

3.2 Experimental results

The PNP plan generated from the POMDP policy for the Escort task was tested in simulation. The tests were performed using Stage[1], using our lab's map as testbed environment. The position and orientation of the human were controlled by the user. To simulate the attention level of the human, we check whether he is looking at the robot or not by using his orientation in the simulation environment.

We have then tested the full framework on a real robot in our lab. Currently, the cameras are able to detect the position, distance and orientation of a person using test patterns. We plan on soon performing further tests using online face tracking techniques.

Through the GUI, the robot first offers to provide assistance and suggests a list of possible destinations. Once the user selects the destination, the Escort Task starts. Whenever the human stops looking at the robot, but is still detected by a camera, the robot will speak to him to draw his attention by inviting him to keep following. When the cameras stop detecting the person, the robot stops its navigation. It then starts turning around hoping to detect the human with its front or rear camera. Once the user is detected again, the robot restarts the navigation module.

The full video of the demonstration, showing the described behaviors, is available on YouTube [2]

4. CONCLUSIONS

We have presented a novel approach to address the problem of ensuring cooperation between a human and a robot agent within a collaborative task.

Our approach makes distinction between Task state variables which pertain to the specific application and Cooperation variables which exclusively deal with the joint intention between agents. While we have described our framework using the Escort Task scenario, adapting the proposed framework in other applications only requires adapting the Cooperation variables to the new task.

Acknowledgments

This work is part of the the COACHES project (Cooperative Autonomous Robots in Complex and Humans Environments) funded by Chist-ERA, 2014 − 2017. [3]

[1] http://playerstage.sourceforge.net/doc/stage-cvs/index.html
[2] `https://youtu.be/qpnZQnTobG8`
[3] `http://www.chistera.eu/projects/coaches`

REFERENCES

[1] Clodic, A., Fleury, S., Alami, R., Chatila, R., Bailly, G., BrÃĺthes, L., Cottret, M., DanÃĺs, P., Dollat, X., EliseÃŕ, F., FerranÃL', I., Herrb, M., Infantes, G., Lemaire, C., Lerasle, F., Manhes, J., Marcoul, P., Menezes, P., and Sabatier, U. P. Rackham: An interactive robot-guide. In *IEEE International Conference on Robot-Machine Interaction (RoMan* (2006).

[2] Ferrari, F., and Mouaddib, A.-I. Hierarchical factored POMDP for joint tasks : application to escort tasks. In *AAAI Fall symposium on Sequential Decisoon-Making for Intelligent Agents (SDMIA 2015)* (Arlington, United States, Nov. 2015), Proceedings of the AAAI Fall symposium on Sequential Decisoon-Making for Intelligent Agents.

[3] Hall, E. T. The hidden dimension, 1966.

[4] Iocchi, L., Jeanpierre, L., Lazaro, M. T., and Mouaddib, A.-I. A practical framework for robust decision-theoretic planning and execution for service robots. In *Twenty-Sixth International Conference on Automated Planning and Scheduling* (2016).

[5] Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. Planning and acting in partially observable stochastic domains. *ARTIFICIAL INTELLIGENCE 101* (1998), 99–134.

[6] Kanda, T., Shiomi, M., Miyashita, Z., Ishiguro, H., and Hagita, N. An affective guide robot in a shopping mall. In *Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction* (New York, NY, USA, 2009), HRI '09, ACM, pp. 173–180.

[7] Matignon, L., Karami, A. B., and Mouaddib, A.-I. A model for verbal and non-verbal human-robot collaboration. In *2010 AAAI Fall Symposium Series* (2010).

[8] Sondik, E. J. The optimal control of partially observable markov processes. Tech. rep., DTIC Document, 1971.

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