# Adaptive Human-Aware Task Planning

**M. Leonetti[1], L. Iocchi[2], Anthony G. Cohn[1], Daniele Nardi[2]**
[1] School of Computing, University of Leeds, UK
[2] DIAG Sapienza University of Rome Italy

## Introduction

Automated planning techniques have been extensively used to generate robot plans in many different scenarios, demonstrating strong generality over properties of particular domains. The distinction in action models between humans and general objects, however, has proved critical in applications that involves people, for instance requiring human-robot interaction, which is a key requirement for *cognitive social* robots.

Therefore, extensions of planning techniques have been considered, which explicitly take into account, and model, human actions and robot actions involving interaction with humans, thus defining a new problem referred to as *human-aware planning* (Sisbot et al. 2007; Cirillo, Karlsson, and Saffiotti 2010). Work in this research line can be grouped into two categories: 1) human-aware path and motion planning, where the main task of the robot is to plan for trajectories in the space taking into account human presence, 2) human-aware task planning (or action planning), where techniques focus on dealing with high-level representations of actions, including interactions (i.e., communication actions between users and robots). In this paper, we will focus on human-aware task planning.

Automatic generation of human-aware plans has been studied using several planning frameworks, including contingent planning under partial observability (Goldhoorn et al. 2018), constraint-based planning with temporal constraints (Köckemann, Pecora, and Karlsson 2014), integer programming (Chakraborti et al. 2016), and model reconciliation (Chakraborti, Sreedharan, and Kambhampati 2018). Iocchi et al. (2016) presented a framework for generating and executing robust plans for service robots in public environments, where the high uncertainty of interacting with non-expert users is tackled by an extended version of Progressive Reasoning Units planning. A different approach based on conditional planning for short-term HRI is addressed by Sanelli et al. (2017), while the use of planning constraints to represent social norms for achieving human-aware plans is proposed by Tomic et al. (2014).



Negotiation of human-robot shared goals has also been addressed in several works. While in most of them such negotiations is performed before plan execution (e.g., (Lallement, de Silva, and Alami 2018)), Sebastiani et al. (2017) extended the Hierarchical Agent-Based Task Planner to resolve uncertainties in goal negotiations on-the-fly, while the plan is being executed, by generating several plans and further merging them through execution variables.

In all approaches mentioned above, every new planning task is treated in isolation, and does not take advantage of previous planning computations, and plan executions. In addition to human-aware planning techniques, adaptive planning techniques have been studied, where learning is used to refine or improve the plans generated by an automated planner. Leonetti, Iocchi, and Stone (2016) introduced a method combining planning and reinforcement learning, which takes advantage of planning to constrain the exploration of the learning agent, and of learning to adapt to the environment, and overcome the limitation of an inaccurate model. Pinto and Fern (2014) proposed a planner that learns a partial policy from previous planning problems in the same domain, so as to guide the search on subsequent instances. Learning heuristics has also been considered (Balduccini 2011; Petrovic and Epstein 2008) within the planning community, as has been learning portfolio planning configurations (Seipp, Sievers, and Hutter 2014). However, pure planning approaches are only concerned with plan genearation, and do not learn from its execution.

In this paper, we discuss the formalization of the problem *Adaptive Human-Aware Task Planning* where Human-Aware Task Planning and Adaptive Planning approaches are properly integrated, including learning from execution. To the best of our knowledge, there has been no previous published research in this direction.

Adaptive Human-Aware Task Planning (AHATP) will be a very important component for cognitive robots interacting with humans as it will allow several problems of current systems to be addressed: 1) lack of adaptivity in user interactions (depending on user profiles, situation context, etc.), 2) poor robustness of high-level plans generated from abstract models, 3) poor integration of social norms and social behaviors in automated plans. An Adaptive Human-Aware Task Planning component will thus enable a social robot to effectively generate and execute plans involving HRI.

We envisage the development of cognitive social robots as systems integrating the following functional modules: 1) Knowledge Representation formalism to describe actions, goals, and plans; 2) Adaptive Goal Selection; 3) Adaptive Human-Aware Task Planner; 4) Adaptive and Robust Plan Execution and Monitoring; 5) Knowledge Transfer.

These five components and the corresponding problems that must be addressed for their development are discussed further in the next sections.

## Knowledge Representation

The design of adaptive cognitive social robots is based on the definition of a proper formalism that will be used to represent all the knowledge needed by the robot to fulfill its tasks. This knowledge contains the following elements: 1) $\mathcal{S}$, a set of states of the environment (including states of users interacting with the robot); 2) $\mathcal{C}$, a set of contexts of the situation (e.g., user profile, situation context); 3) $\mathcal{A}$, the actions available; 4) $\mathcal{G}$, a set of goals; 5) $\mathcal{P}$, a set of plans/behaviors/policies, 6) $\mathcal{E}$, a set of past experiences. Note that we distinguish states from contexts depending on whether the robot can affect their evolution (states) or not (contexts). Therefore, only state variables appears in the post-conditions of actions.

A knowledge base can be thus represented as

$$\mathcal{KB} = \langle \mathcal{S}, \mathcal{C}, \mathcal{A}, \mathcal{G}, \mathcal{P}, \mathcal{E} \rangle$$

and a proper representation language must be defined for each of these elements.

At a given instant of time $t$, the agent will have a particular instance of knowledge $K_t$ about the world that can be represented as

$$K_t = \langle S_t \in \mathcal{S}, C_t \in \mathcal{C}, A_t \subseteq \mathcal{A}, G_t \subseteq \mathcal{G}, P_t \in \mathcal{P}, E_t \subseteq \mathcal{E} \rangle$$

where $S_t \in \mathcal{S}$ is the current state (as known by the agent), $C_t \in \mathcal{C}$ are the current contexts, $A_t \subseteq \mathcal{A}$ are the actions available at this time, $G_t \subseteq \mathcal{G}$ are the current goals, $P_t \in \mathcal{P}$ is the current plan, $E_t \subseteq \mathcal{E}$ is the experience collected so far.

The explicit representation of the above mentioned elements allows for a formal definition of the problems to be addressed in the functional modules identified above. Such problem definitions are briefly summarized here and discussed in more details later on.

**Adaptive Goal Selection**. Given the current knowledge $K_t$, compute (possibly new) goals $G_{t+1}$, according to the current state, current contexts and experience.

**Adaptive Human-Aware Task Planning**. Given the current knowledge $K_t$, compute a (possibly new) plan $P_{t+1}$, according to the current state, current contexts, goals, and experience.

**Plan Execution and Monitoring**. Given the current knowledge $K_t$, execute the current plan monitoring state and context evolution $S_{t+1}$, $C_{t+1}$ and collecting new experience $E_{t+1}$.

**Knowledge Transfer**. Given the current knowledge $K_t$, exploit past experience to derive (possibly new) contexts $C_{t+1}$.

## Adaptive Goal Selection

As social cognitive robots are involved in dynamic and unpredictable scenarios, while they continuously perceive the environment, monitor the current situation, and execute the current plan to achieve the current goal, new pieces of knowledge may bring to an opportunity or a necessity to switch to a different goal. As described in the previous section, the robot's $\mathcal{KB}$ contains a set of goals $\mathcal{G}$ that the robot has to accomplish and the current instance of knowledge $K_t$ can be used to determine which goals are suitable at this moment, thus computing $G_{t+1}$.

This functionality is implemented as a goal selection procedure based on reasoning about the current knowledge of the robot. More specifically, such a reasoning module will exploit the information represented in the current knowledge base $K_t$ to compute the best achievable goal. Experience contained in the knowledge base will also be used for adaptive goal selection based on the context (e.g., user profiles, special situations, etc.).

The selection of the *best* goals will be based on the definition of a *utility* (either numeric or logic-based) that is a key component of the adaptive goal selection module.

## Adaptive Human-Aware Task Planning

High-level plans drive the behaviour of the robot at the highest (symbolic) control level, including the interaction with the users. The behaviour of the robot must evolve over time, by proposing new ways of achieving the goal based on the success of previous executions. The system takes advantage of automated reasoning, to leverage an approximate model and propose actions that lead to the goal in ways that are potentially more successful than before, and machine learning to assess the effectiveness of the executed actions and optimize the behaviour over time.

Generating behaviours for tasks that involve humans is inherently difficult, since the two strategies most employed in the planning and learning literature cannot be applied satisfactorily: create compact models for planning or build reliable simulations for learning. However, long-term deployment offers an opportunity to address this challenge, by optimizing the behaviour over time through well-defined exploratory actions, limited in scope by the rationality of an automated reasoner, and the guidance of an approximate model. With each planning session, the planner will contribute a new alternative, eventually building up behaviour over several executions. Throughout the learning process, the model can be adapted, if the system accumulates evidence of incorrectness. However, no amount of model learning will make it perfect, which also motives the use of model-free adaptation techniques, so as to complement an imperfect model.

## Adaptive and Robust Plan Execution and Monitoring

Plans generated by task planners still need to be refined in order to be effective in the real world. More specifically, it is necessary to increase their robustness in real situations, where unmodelled or incorrectly modelled features may invalidate the choices made at planning time. Moreover, an effective plan for robots interacting with people must support complex features such as time-durative actions, on-line sensing, parallel execution, joint actions, interrupts, etc. To this end, an expressive plan representation formalism is needed.

Moreover, the robot should be equipped with a robust plan execution layer that processes all perception inputs at run-time, including the ones coming from human interactions, in order to allow the robot to update its state and effectively and efficiently achieve its goals with minimal replanning and failures. The robust plan is a modified version of the nominal plan that reaches the same goals, but it is more reliable in situations not modelled at planning time and thus can reach the goals in more actual cases than the nominal plan.

Finally, the execution and monitoring component can also be configured in order to define different operation modes (i.e., different ways of executing the same plan) and to collect past experience on the use of such operation modes. Such experience will allow for adaptive operation mode selection according to the context (e.g., user profile and therapy) that will thus enable personalized plan execution.

From our past experience in developing complex robot and multi-robot applications using planning and plan execution techniques, we propose the use of Petri Net Plans (PNP)[1] (Ziparo et al. 2011) to exploit the high representation power of Petri Nets and its effectiveness in representing high-level robot plans. The PNP formalism is suitable to represent time-durative actions, on-line sensing, parallel execution, joint actions, interrupts, as well as multi-robot operators that can be useful also for plans involving human-robot interaction. Moreover, PNP has been already successfully used in planning and plan execution for HRI applications, in particular to define social norms (Nardi and Iocchi 2014) and to use conditional planning to generate short interactions (Sanelli et al. 2017).

When using the PNP formalism, generation of robust plans can be obtained through the use of execution rules that can be either provided by the system designer (Iocchi et al. 2016) or learned from experience or by demonstration.

[1]http://pnp.diag.uniroma1.it

## Knowledge Transfer

A fundamental aspect of our proposal for Adaptive Human-Aware Task Planning is that the system takes advantage of experience acquired *during execution*. Past experience needs, therefore, to be condensed so as to provide context and initial information for the adaptive planner and the execution monitor. Learning must be performed in the real world, rather than in simulation as is common in robotics, because of the inherent difficulty in correctly simulating tasks and environments involving human presence, as previously mentioned. Therefore, it is particularly critical that appropriate models are leveraged to constrain the exploration, and that the system generalizes efficiently across similar executions, so as to reuse as much knowledge as possible.

Execution similarity may be modelled at two levels: static and dynamic. Static features assess the similarity between the elements of a situation that do not depend on the robot's behaviour, such as the initial state of the environment $S_0$ and the contextual information $C_0$. Dynamic features determine the similarity between behaviours that have proven successful, regardless of static features, that is, in potentially different situations. Ultimately, the system must learn contexts in which a given successful behaviour could be transferred to the current context, regardless of apparently different situations. For instance, if similar behaviours are successful different contexts, experience between the two can be merged, even if the knowledge transfer would not have taken place looking at static features only. Note that learning the features entirely with recent deep learning techniques is likely to be unfeasible, as we cannot expect to be able to collect enough real-world data. Therefore, we expect transfer to be based on statistical methods, which can control for the uncertainty brought about by the limited information. The features are used to cluster execution contexts, and determine executions between which knowledge can be shared.

## Conclusions

The goal of deploying adaptive cognitive social robots in public environments requires addressing many interesting scientific problems at the intersection between artificial intelligence (in particular, knowledge representation, automated planning and machine learning) and robotics. In particular, we believe that such social robots may significantly benefit from the use of a proper integration of automated planning and machine learning techniques.

In this abstract, we have defined a set of problems that should be solved in order to achieve the above mentioned objective. In particular, we advocate the development of adaptive human-aware task planning techniques that will certainly be the basis of a future generation of cognitive social robots.

## References


Balduccini, M. 2011. Learning and using domain-specific heuristics in asp solvers. *AI Communications* 24(2):147–164.

Chakraborti, T.; Zhang, Y.; Smith, D. E.; and Kambhampati, S. 2016. Planning with resource conflicts in human-robot cohabitation. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, 1069–1077. International Foundation for Autonomous Agents and Multiagent Systems.

Chakraborti, T.; Sreedharan, S.; and Kambhampati, S. 2018. Human-aware planning revisited: A tale of three models. In *IJCAI-ECAI XAI/ICAPS XAIP Workshops*.

Cirillo, M.; Karlsson, L.; and Saffiotti, A. 2010. Human-aware task planning: An application to mobile robots. *ACM TIST* 1(2):15:1–15:26.

Goldhoorn, A.; Garrell, A.; Alquézar, R.; and Sanfeliu, A. 2018. Searching and tracking people with cooperative mobile robots. *Autonomous Robots* 42(4):739–759.

Iocchi, L.; Jeanpierre, L.; Lazaro, M. T.; and Mouaddib, A. 2016. A practical framework for robust decision-theoretic planning and execution for service robots. In *Proceedings of the Twenty-Sixth International Conference on Automated Planning and Scheduling, ICAPS 2016, London, UK, June 12-17, 2016.*, 486–494.

Köckemann, U.; Pecora, F.; and Karlsson, L. 2014. Grandpa hates robots-interaction constraints for planning in inhabited environments. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Lallement, R.; de Silva, L.; and Alami, R. 2018. HATP: hierarchical agent-based task planner. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, 1823–1825.

Leonetti, M.; Iocchi, L.; and Stone, P. 2016. A synthesis of automated planning and reinforcement learning for efficient, robust decision-making. *Artificial Intelligence* 241:103–130.

Nardi, L., and Iocchi, L. 2014. Representation and execution of social plans through human-robot collaboration. In *Social Robotics - 6th International Conference, ICSR 2014, Sydney, NSW, Australia, October 27-29, 2014. Proceedings*, 266–275.

Petrovic, S., and Epstein, S. L. 2008. Random subsets support learning a mixture of heuristics. *International Journal on Artificial Intelligence Tools* 17(03):501–520.

Pinto, J., and Fern, A. 2014. Learning partial policies to speedup mdp tree search. In *UAI*, 672–681. Citeseer.

Sanelli, V.; Cashmore, M.; Magazzeni, D.; and Iocchi, L. 2017. Short-term human-robot interaction through conditional planning and execution. In *Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling, ICAPS 2017, Pittsburgh, Pennsylvania, USA, June 18-23, 2017.*, 540–548.

Sebastiani, E.; Lallement, R.; Alami, R.; and Iocchi, L. 2017. Dealing with on-line human-robot negotiations in hierarchical agent-based task planner. In *Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling, ICAPS 2017, Pittsburgh, Pennsylvania, USA, June 18-23, 2017.*, 549–557.

Seipp, J.; Sievers, S.; and Hutter, F. 2014. Fast downward cedalion. *International Planning Competition Planning and Learning Part: planner abstracts*.

Sisbot, E. A.; Marin-Urias, L. F.; Alami, R.; and Simeon, T. 2007. A human aware mobile robot motion planner. *IEEE Transactions on Robotics* 23(5):874–883.

Tomic, S.; Pecora, F.; ; and Saffiotti, A. 2014. Too cool for school - adding social constraints in human aware planning. In *Proc. of 9th International Workshop on Cognitive Robotics*.

Ziparo, V. A.; Iocchi, L.; Lima, P. U.; Nardi, D.; and Palamara, P. F. 2011. Petri net plans - A framework for collaboration and coordination in multi-robot systems. *Autonomous Agents and Multi-Agent Systems* 23(3):344–383.