Planning Interactions as an Event Handling Solution for Successful and Balanced Human-Robot Collaboration

Silvia Izquierdo-Badiola[1,2], Carlos Rizzo[1] and Guillem Alenyà[2]

***Abstract*— Dealing with the stochastic nature of human behaviour in Human-Robot Collaboration (HRC) remains a well known challenge that needs to be tackled. Automated task planning techniques have been implemented in order to share the workload between the agents, but these still lack the necessary adaptability for real-world applications. In this paper, we extend our previous work presented in [1], where an improved task planning framework integrating an agent model was presented, anticipating and avoiding failures in HRC by reallocating the actions in the plan based on the agents' states. This work introduces the integration of interaction actions into the planning framework, in order to deal with situations where the issue reflected by a change in an agent state might be better handled with an interaction between the agents than by an action reallocation. Preliminary evaluation shows promising results of how this framework can help to increase the success in HRC plans, as well as the balance in workload distribution between the agents, which constitutes a key element in a collaboration.**

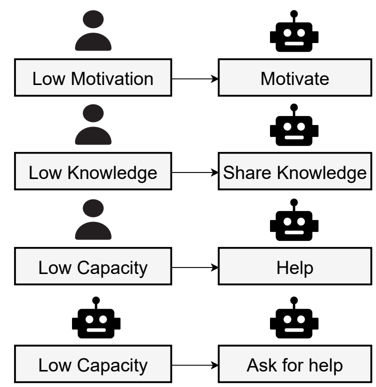

Fig. 1. Inference of the type of interaction required (on the right), based on the agent states (on the left).

I. INTRODUCTION

The topic of Human-Robot Collaboration (HRC) has gained substantial attention over the past years, with the idea of combining strengths from humans and robots to improve efficiency and productivity in shared plans. However, there are still several challenges that need to be tackled when developing such systems for real-world successful applications.

AI (or automated) task planning frameworks have been implemented to generate and distribute the sequence of actions required to achieve a shared goal, when given a world model (domain) with an initial state and a high-level goal. In our previous work [1], the failure of these frameworks to deal with the unpredictable nature of the human behaviour was recognised as one of the main challenges in HRC. We targeted the gap between agent state modelling and AI task planning in order to improve adaptability. An AI planning framework for HRC plans integrating the agents' states as action costs was developed, in order to anticipate and avoid failures due to unforeseen human behaviour. The agent state was defined in terms of three components: *capacity*, *motivation* and *knowledge*. These three elements are believed to capture unexpected events, covering the main reasons why an agent would not contribute towards a shared goal. The system continuously checks if the agent is physically capable, focused, and has all the required knowledge to complete the shared task. A change in the agent state during the plan execution would be dealt with by replanning, taking into account the new state in the action costs and accordingly reallocating the actions in the plan, avoiding the failures that would have occurred if assigned to this agent. This, although successful in terms of failure prevention, would sometimes lead to an unbalanced contribution between the agents.

This paper extends this work by adding Human-Robot Interaction (HRI) capabilities into the plan definition and generation. We introduce a new action type to trigger interactions, which is useful for the cases where an interaction might help to get the agent to collaborate again. Now, the system can choose among reallocating actions or trying to interact to solve the issue (lack of motivation, knowledge or capacity) (Fig. 1). We intend to answer the following research questions. In case of unexpected events affecting the agents' contribution in a collaboration,

- **R1 -** Can we maintain the balance in workload allocation by integrating interactions in a planning framework based on an agent model?
- **R2 -** Can this help to mitigate failures and increase the success rate?

The contribution of this work consists in a modelling and planning framework that plans for both agents in a collaboration and integrates the right interactions between the agents' based on their states. Our hypothesis is that in this way, unplanned events reflected in the agent behaviour can be better dealt with, and a more successful and balanced

[1]Eurecat, Centre Tecnològic de Catalunya, Robotics and Automation Unit, Barcelona, Spain {silvia.izquierdo, carlos.rizzo}@eurecat.org

[2]Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, Spain {sizquierdo,galenya}@iri.upc.edu

*S. Izquierdo is a fellow of Eurecat's *Vicente López* PhD grant program. This work was partially financed by MCIN/ AEI /10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR" under the project ROB-IN (PLEC2021-007859) and by the Catalan Government through the funding grant ACCIÓ-Eurecat.

collaboration can be achieved.

II. Related Work

Previous research has dealt with the combination of task planning and HRI in a number of ways. A number of works focus on modelling the interaction through planning formalisms. In [2], a general PDDL planning domain for the formalisation of HRI is introduced. Sanelli *et al.* [3] define a conditional planning domain to generate and execute short-term interactions by a robot deployed in a public environment. The authors in [4] deviate from PDDL, developing a planner based on graph models for HRI domains. In [5], the human-robot interactions are included in the planning system by using a complex taxonomy of preferences [6], and in [7], this approach is evaluated in assistive robotic applications. Also in the assistive domain, in [8], the authors tackle the problem of agile and effective interactions by using a learned policy using a simulator [9]. These works concentrate the effort on modelling and planning for interactions, but do not integrate them into a high level collaborative plan where a joint goal needs to be achieved by the agents, which is the case targeted in this work.

In [10], the feasibility and cost of verbal communication is taken into account at the task planning level for a task where the robot issues commands to the human partner in order to achieve a goal. In [11], the authors propose a solution for human-robot collaborative plans where the robot interacts in an adaptive way based on the human partner's level of knowledge about the task. In this work, the interactions are not meant to handle unexpected events or modify the plan, but to provide explanations. Devin and Alami [12] explicitly use an estimation of the human knowledge in the shared plan execution, focusing on producing a less intrusive behaviour of the robot. Theory of mind is used to detect when verbal communication is required to solve a knowledge divergence between the agents. Our work elaborates on this idea, covering, in addition to knowledge, the elements of capacity and motivation of the agents.

To the best of our knowledge, there is no work focusing on planning for interactions that directly handle an issue fully reflected in an agent state model in a collaboration, with the aim of both avoiding failures and maintaining the workload share as balanced as possible between the agents.

III. Methodology

The workflow we propose in order to improve HRC plans through planning for interactions can be seen in Fig. 2. The planning system has been implemented in a standardised environment using the Planning Domain Definition Language (PDDL) [13] and the modular ROSPlan framework [14]. This allows for the concept to be easily extended to a variety of scenarios and applications involving Human-Robot Collaboration. The initial collaborative plan assigns a sequence of actions to each agent, assuming both agents have full *motivation*, *knowledge* and *capacity* to complete their assigned part of the plan. A change in these states, which are constantly monitored during the plan execution,

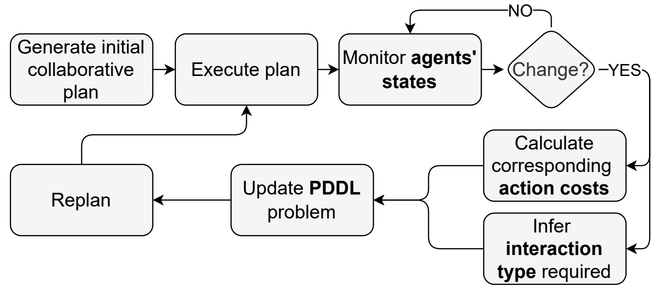

Fig. 2. Workflow implemented in order to adapt to changes in the agents' states during a collaboration, by planning for both the required interactions and the reallocation of actions between the agents.

will stop the current plan execution and trigger a replan. As these states are represented in terms of action costs in the PDDL plan definition (as described in our previous work [1]), the new plan will take them into consideration, reassigning the actions between the agents as needed and avoiding action failure due to the agent state. The novelty we introduce here is that the agents' states are also used to infer a possible interaction that could handle unexpected events and avoid some actions having to be reassigned to a different agent, maintaining a more balanced workload share in the collaboration. The PDDL problem is also updated with this information, and the new plan will include the required interactions.

The inference of which interaction action is required is based on the agents' states as shown in Fig. 1. Four types of interactions are introduced: motivate, ask for help, help, and share knowledge. In terms of actions, two types are defined in the PDDL domain: the execution actions, contributing to the plan completion, and the interaction actions. The interaction actions are expected to handle and avoid action failures that would have been caused by the lack of *motivation*, *capacity* or *knowledge* in a participating agent. They have the effect of decreasing the execution action costs reflecting the relevant component of the agent state (Fig. 3). The change of action cost by the interaction action will take effect during the plan execution, but is taken into account by the planner at the plan generation stage.

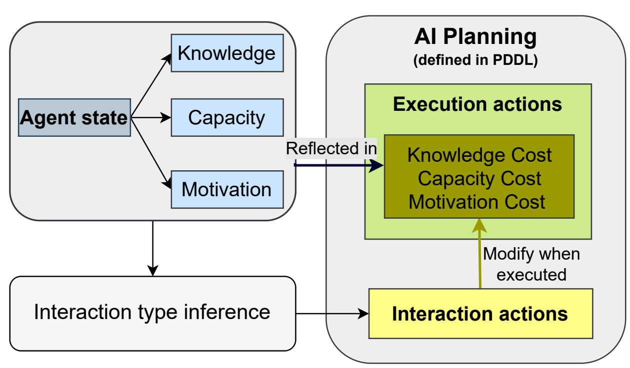

Fig. 3. Block diagram representing the integration of the agents' states and the derived interaction actions required into the AI task planning framework.

PDDL problem	
predicate	**objects**
interaction_not_needed	/
interaction_needed	interaction_type, agentfrom, agentto

PDDL domain		
	execution	**interaction**
actions	• move • place • grasp	• motivate • help • shareknowledge • askforhelp
pre-condition	`interaction_not_needed`	`interaction_needed ?interaction_type ?agentfrom ?agentto`
effect	`increase (`**`total-cost`**`) (+(+(+ (`**`knowledge`**` ?agent) (`**`motivation`**` ?agent) (`**`capacity`**` ?agent) (specific_action_cost ?agent)`	Each interaction action deals with a different element from the agent model. The motivate action sets the motivation cost (increased from a previous distraction) to 0: `assign (`**`motivation`**` ?human) 0 interaction_not_needed not (interaction_needed ?motivate ?robot ?human)`

Fig. 4. Integration of interaction capabilities into the AI task planning framework, defined using PDDL.

Fig. 4 provides more detail on how the interactions are integrated in the planning PDDL definition. The execution actions increase the plan cost based on the *motivation*, *capacity* and *knowledge* costs of the agent that each action is assigned to. The execution actions contain the precondition that an interaction is not needed, whilst the interaction actions have this precondition as an effect. Each interaction action deals with a different element from the agent model, setting the cost related to this element to the minimum. As an example, if low motivation (high motivation cost) is detected on the human, a motivate action from the robot might be required. This is reflected in the PDDL problem by setting the predicates *(not(interaction_not_needed))* and *(interaction_needed ?motivate ?robot ?human)*. The motivate action will therefore be assigned to the robot, lowering the human motivation cost to 0, and the remaining actions can be distributed again between the agents. Before the introduction of the interaction actions, the human motivation cost would have remained high, and all of the actions in the plan would have been assigned to the robot. In this work, we assume the interaction actions are successful and have the desired effect, solving the relevant *motivation*, *capacity* or *knowledge* issue.

IV. Preliminary Results

We present three cases where the collaboration is improved by the integration of interaction actions in the collaborative planning framework, as a way of dealing with unexpected events due to the agents. Our example consists of a robot and a human that need to move boxes between different locations (Fig. 5). The specified goal is for the green box to be moved from *wp3* to *wp2* and the blue box to be moved from *wp4* to *wp5*. Initially, *motivation*, *capacity* and *knowledge* costs are set to a low value, meaning both agents are capable, focused, and have all the required knowledge to complete the shared task, which produces the plan in Fig. 6 - InitialPlan.

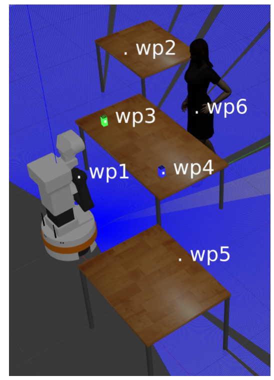

Fig. 5. Simulated scenario for evaluation consisting of two agents (robot and human) that have to transport blocks to different locations.

Case 1) The human agent gets distracted at the start of the plan execution. This is reflected in the motivation component of the human state, which is in turn reflected in a very high human motivation cost affecting all human actions. Without the possibility of interacting, all the actions in the plan would have been assigned to the robot in order to avoid failures. This, however, defeats the purpose of the collaboration. With the integration of the interaction actions, the problem is solved by introducing a *motivate* action from the robot to the human, which will lower the human motivation cost, allowing for the workload balance to be recovered for the rest of the plan (Fig. 6 - Case1).

Case 2) The robot needs help to move after grasping the block as the path is blocked. This is reflected in the capacity component of the robot state. Observe that a replan without interactions will lead to the same failing initial plan. As the robot has already picked the block, placing it back and moving away (place, then move) involves the same cost as the initial plan (move, then place). By introducing the interaction domain, a need for the *askforhelp* action is recognised by the system, which will be reflected in the PDDL domain. The replan will include an *askforhelp* action from the robot to the human, followed by a *help* action from the human to the robot. After this, the path is assumed to be unblocked, setting the robot capacity cost back to 0. The rest of the plan remains unchanged, reaching the goal whilst respecting the intended workload balance (Fig. 6 - Case2).

Case 3) The human doesn't know where the placing waypoint is. This time the event is reflected in the knowledge component of the human. Before the implementation of the interaction framework, as the human has already picked the block, the same failing plan is generated. If the need for a *shareknowledge* action is detected, the new plan will contain this action from the robot to the human, lowering the human knowledge cost and maintaining the initial plan. The desired goal can therefore be reached whilst maintaining a balanced collaboration (Fig. 6 - Case3).

V. Conclusions and Future Work

In this paper, we have improved and extended an AI task planning system that anticipates and prevents failures in HRC plans, by considering the agents' states to replan ahead. Depending on this state, the replan might involve the reassignment of the actions or an interaction between the agents, maintaining a balanced collaboration whilst avoiding action failure.

The agent state is represented in terms of *capacity*, *knowledge*, and *motivation*, and is constantly monitored during the plan execution. We define a PDDL domain including two types of actions: plan execution actions and interaction actions. The agents' states are both represented as execution action costs in the PDDL domain, and used to make the decision on the need of an interaction. The interaction actions deal with a failure that would have been caused by the lack of *capacity*, *knowledge* or *motivation* in an agent, and have the effect of reducing the part of the execution action costs representing this element. In this way, a change in an agent state, which triggers a replan, can be handled not only by the reassignment of actions in the generated plan, but also by interacting, which might sometimes be preferred to maintain a balanced collaboration.

The system can be easily expanded, implemented in a standardised environment using PDDL and the modular ROSPlan framework. The effect of the introduction of interactions into the planning framework on the collaboration was analysed for three cases by means of simulations. From the preliminary results, we can affirmatively answer the research questions posed regarding the human-robot collaboration, in terms of workload allocation improvement and failure mitigation through the addition of interaction actions.

As future work, we envisage to extend the evaluation to a significant number of runs and scenarios, including metrics such as success rate and workload balance allocation. We also intend to improve and develop the interaction inference block based on the agent state. In this work, we associate a type of interaction to a single failing element in the agent model. We will need to deal with the cases where several elements in the agent model are low (e.g. agent has both a low motivation and capacity): Which interaction(s) are needed? Which interaction to trigger first? Furthermore, this work assumes the interactions are always successful at dealing with the detected issue in the collaboration. However, a way of dealing with the cases where the interaction fails will need to be investigated. Finally, a more detailed definition of what the interaction type involves will be required for the implementation of the framework in a real system.

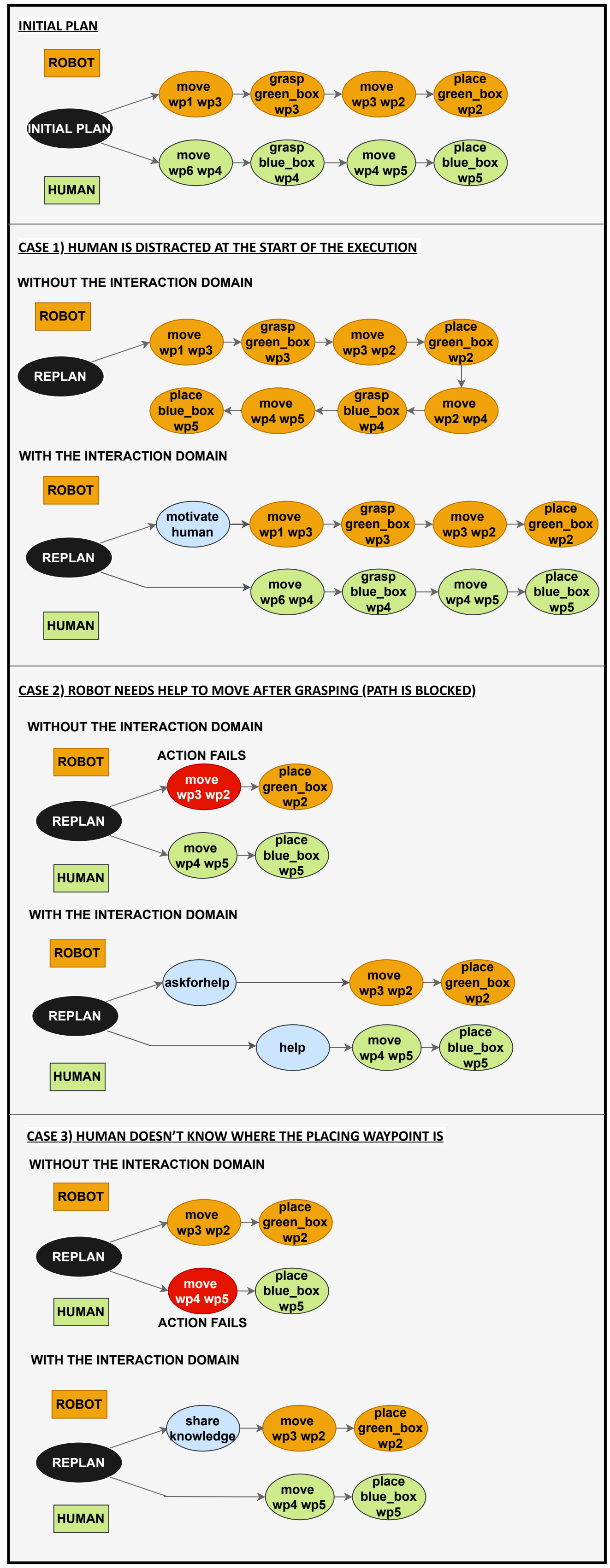

Fig. 6. Three cases presenting the benefits of the integration of interaction capabilities into the planning component to increase the success rate and workload allocation between the agents in a collaboration.

References

[1] S. Izquierdo-Badiola, G. Canal, C. Rizzo, and G. Alenyà;, "Improved task planning through failure anticipation in human-robot collaboration," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 7875–7880.

[2] R. Fuentetaja, A. García-Olaya, J. García, J. C. González, and F. Fernández, "An automated planning model for hri: Use cases on social assistive robotics," *Sensors*, vol. 20, no. 22, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/22/6520

[3] V. Sanelli, M. Cashmore, D. Magazzeni, and L. Iocchi, "Short-term human-robot interaction through conditional planning and execution," in *ICAPS*, 2017.

[4] L. Manso, P. Bustos, R. Alami, and G. Milliezf, "Planning human-robot interaction tasks using graph models," in *International Workshop on Recognition and Action for Scene Understanding*, Oct. 2015, pp. 195–208, © 2015 The Authors.

[5] G. Canal, G. Alenyà, and C. Torras, "Adapting robot task planning to user preferences: an assistive shoe dressing example," *Autonomous Robots*, vol. 43, no. 6, pp. 1343–1356, 2019.

[6] G.Canal, G. Alenyà, and C. Torras, "A taxonomy of preferences for physically assistive robots," in *IEEE International Symposium on Robot and Human Interactive Communication*, 2017, pp. 292–297.

[7] G. Canal, C. Torras, and G. Alenyà, "Are preferences useful for better assistance?: A physically assistive robotics user study," *ACM Transactions on Human-Robot Interaction*, vol. 10, no. 4, pp. 1–19, 2021.

[8] A. Andriella, C. Torras, C. Abdelnour, and G. Alenyà, "Introducing caresser: A framework for in situ learning robot social assistance from expert knowledge and demonstrations," *User Modeling and User-Adapted Interaction*, 2022. [Online]. Available: http://www.iri.upc.edu/publications/show/2580

[9] A. Andriella, C.Torras, and G. Alenyà, "Learning robot policies using a high-level abstraction persona-behaviour simulator," in *IEEE International Symposium on Robot and Human Interactive Communication*, 2019.

[10] G. Buisan, G. Sarthou, and R. Alami, "Human aware task planning using verbal communication feasibility and costs," in *Social Robotics*, A. R. Wagner, D. Feil-Seifer, K. S. Haring, S. Rossi, T. Williams, H. He, and S. Sam Ge, Eds. Cham: Springer International Publishing, 2020, pp. 554–565.

[11] G. Milliez, R. Lallement, M. Fiore, and R. Alami, "Using human knowledge awareness to adapt collaborative plan generation, explanation and monitoring," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016, pp. 43–50.

[12] S. Devin and R. Alami, "An implemented theory of mind to improve human-robot shared plans execution," *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 319–326, 2016.

[13] M. Fox and D. Long, "PDDL2.1: An extension to PDDL for expressing temporal planning domains," *Journal of artificial intelligence research*, vol. 20, pp. 61–124, 2003.

[14] M. Cashmore, M. Fox, D. Long, D. Magazzeni, B. Ridder, A. Carrera, N. Palomeras, N. Hurtós, and M. Carreras, "ROSPlan: Planning in the robot operating system," *Proceedings International Conference on Automated Planning and Scheduling, ICAPS*, pp. 333–341, 2015.