To Lead or to Follow? Adaptive Robot Task Planning in Human-Robot Collaboration

Ali Noormohammadi-Asl, *Member, IEEE*, Stephen L. Smith, *Senior Member, IEEE*, and Kerstin Dautenhahn, *Fellow, IEEE*

***Abstract*—Adaptive task planning is fundamental to ensuring effective and seamless human-robot collaboration. This paper introduces a robot task planning framework that takes into account both human leading/following preferences and performance, specifically focusing on task allocation and scheduling in collaborative settings. We present a proactive task allocation approach with three primary objectives: enhancing team performance, incorporating human preferences, and upholding a positive human perception of the robot and the collaborative experience. Through a user study, involving an autonomous mobile manipulator robot working alongside participants in a collaborative scenario, we confirm that the task planning framework successfully attains all three intended goals, thereby contributing to the advancement of adaptive task planning in human-robot collaboration. This paper mainly focuses on the first two objectives, and we discuss the third objective, participants' perception of the robot, tasks, and collaboration in a companion paper.**

***Index Terms*—Human-robot collaboration, adaptive task planning, proactive task allocation, human preference and performance.**

I. Introduction

COBOTS, short for collaborative robots, have signified a transformative leap from traditional industrial robots, working isolated from humans, to robots that can share their workspace with their human coworkers, laying the ground to exploit the synergy of human-robot collaboration (HRC). Although cobots are currently slower and less powerful than traditional industrial robots, mainly due to their proximity to humans and safety concerns, they are easy to install and relocate and are productive and cost-effective automation solutions for diverse work environments, even small enterprises [1]. Leveraging these capabilities and establishing a seamless collaboration, however, can be achieved only through human-aware programming of cobots [2], empowering them to learn, adapt, and work robustly.

One of the main challenges in programming cobots is enabling them to adapt to their human teammate, especially their preferences. This topic has been extensively researched in the context of human-robot interaction (HRI) and collaboration, primarily centering on considering human preferences and enhancing human satisfaction and perception of the robot [3]–[6]. Clearly, a high level of human perception of the robot facilitates effective long-term collaboration between humans and cobots, but a question remains: "Do only the human teammate's satisfaction and perception matter?". Cobots are typically less expensive compared to traditional industrial robots, yet they still need to demonstrate sufficient productivity to convince business owners to invest in them [2]. Hence, team performance, in addition to human perception of the robot and collaboration, are two important factors to consider in cobot programming.

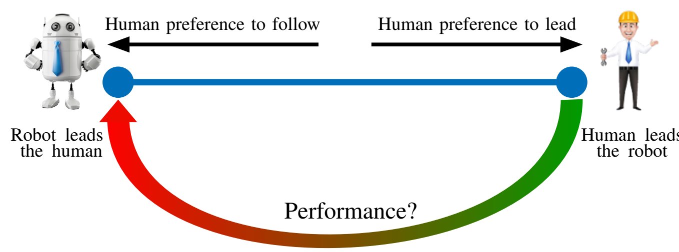


Fig. 1. Spectrum of leading/following roles in human-robot collaboration

Both the human and robot contribute to team performance with their complementary skill, but in the context of task planning and scheduling, the robot's high computational and planning abilities allow it to take a more significant and leading role, ensuring good team performance in the short term. However, due to human presence and dynamic environment uncertainties, it is sometimes more efficient for the human interaction partner to plan for the team [2]. However, the objectives of maximizing team performance and human perception may be conflicting. This leads to the following problem, which is the focus of this paper: *How do we enable a cobot to adapt to its human coworker's preferences to optimize human perception while keeping the team performance at an acceptable level?*

We previously conducted an online user study involving a scenario with a single human and a robot, requiring the two parties to collaborate to accomplish a given task [7]. We considered three robot strategies: prioritizing the human (Strategy 1), prioritizing the robot (Strategy 2), and balancing both (Strategy 3). Based on the results, Strategies 1 and 3 enhanced human perception of the collaboration compared to

This research was undertaken, in part, thanks to funding from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada 150 Research Chairs Program.

Stephen L. Smith and Kerstin Dautenhahn are with the Department of Electrical and Computer Engineering, Faculty of Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: stephen.smith@uwaterloo.ca, kerstin.dautenhahn@uwaterloo.ca).

Ali Noormohammadi Asl is with the Department of Systems Design Engineering, Faculty of Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: ali.asl@uwaterloo.ca).

Strategy 2, with no significant difference between Strategies 1 and 3. Those previous findings proposed that having a balanced robot plan in collaborative scenarios benefits the team without diminishing human perception of the collaboration. In this prior work, although the robot's decisions were based on the state of the task, there was no adaptation to human preferences, and only fixed strategies were considered.

Resting on the insights gathered from this initial study, we have proposed a framework that equips the robot to estimate and adapt its planning to

1) the human leading/following preferences, and
2) the human performance.

In this framework, in contrast to most common approaches, which strictly focus on one end of the leading/following spectrum, the cobot gradually and continuously adapts to human performance and (leading/following) preferences in pursuit of long-term performance and can encompass the entire spectrum between acting solely as a leader or solely a follower. Fig. 1 encapsulates the main idea of this framework.

The framework allows the robot to monitor and adapt to changes in human preferences and performance during the collaboration rather than relying on fixed, long-term preferences and performance. For example, consider a person with a following preference who becomes fatigued at some point and then prefers to assign more tasks to the robot, or a scenario in which a person with a high level of performance becomes confused and makes a few mistakes. This framework can be applied to different collaborative scenarios; however, our research focuses specifically on task selection/allocation and scheduling in human-robot collaboration.

In a basic task allocation problem, agents need to be assigned tasks with associated payoffs, aiming to optimize overall team payoff. This is a widely explored topic in multi-robot teams [8], as well as HRC, involving a blend of human and robot agents [9]–[13]. Our work focuses on a single-human, single-robot teams, and what makes it different from conventional task allocation systems is: *the human and cobot's agency in selecting their own tasks and assigning tasks to each other instead of being assigned merely by one of the agents or a third party (e.g., a manager/central controller).*

The provided agency allows the human to demonstrate and implement their leading/following preference in their collaboration with the cobot. Subsequently, throughout the collaboration, the cobot needs to estimate the latent human preference, monitor and estimate the human's performance, and adapt its planning online. The robot performs two-step task planning at each step: first, task allocation considering its belief about human preference and performance, and second, task scheduling. In this framework, the robot needs to reassume the leading role when the human's performance is poor, even if the human agent prefers to lead the team.

In [14], we tested the framework in a simulation environment using a simplified model of human decision-making. In [15], we also implemented it on an actual mobile manipulator, the Fetch robot, and tested it by having the experimenter enact some possible different human collaboration styles. However, to test the system and planning framework's effectiveness and their influence on human perception of the robot, we conducted a user study, which is the primary focus of this paper. We consider a collaborative scenario, inspired by the kitting task, with a set of precedence-constrained tasks that must be completed through collaboration between the human and the cobot. Both agents make their decisions asynchronously and can select tasks for themselves or assign them to each other.

A. Contributions

Restating the problem that we aim to tackle in this paper, adapting to human leading/following preferences while maintaining team performance at a high level, the main contributions of this paper are as follows:

1) We propose a robot planning framework enabling the robot to consider both human leading/following preference and performance simultaneously. We apply this framework specifically to the task allocation problem and present a two-step planning structure: 1- Task allocation by considering the robot's belief about the human agent's performance and leading/following preference, and 2- task scheduling.
2) Our planning algorithm dynamically updates task states based on both agents' actions and actively identifies and addresses errors made by the human agent.
3) We present the development and practical implementation of adaptive robot task planning in a collaborative scenario involving a robot performing autonomous pick-and-place.
4) Through a comprehensive user study involving 48 participants, we demonstrate that the planning framework empowers the robot to proactively infer participants' performance and leading/following preferences in its task planning. The study also reveals the framework's ability to adapt to changes, such as participants preferring to follow the robot in challenging tasks or the robot reassuming the leading role when participants' performance decreases.

Preliminary versions of parts of this work appeared in the conference papers [14], [15]. In [14] we introduced a planning framework based on human preference and performance, limited to a collaborative scenario in a simulation environment with a simplified human decision-making model. In contrast, this paper applies the framework to a more complex collaborative scenario involving an actual robot working alongside recruited participants. In [15] we presented an initial version of implementing this collaborative scenario and assessed the planning framework's performance for four different scenarios conducted by the experimenter. However, the current paper presents the final version of the experimental setup with modifications to the planning framework to minimize frequent changes in robot planning. Importantly, it evaluates the efficiency of the proposed planning framework for an autonomous robot collaborating individually with each of the 48 participants (involving in total 144 tasks) and discusses specific cases to demonstrate the framework's adaptability to different participant preferences and performance levels and their variation.

We also note that due to the broad scope of the planning framework and user study, we have written this paper to focus primarily on aspects of robot planning, both theory and user study evaluation. We have then written a companion paper [16], to focus on participants' perceptions of the robot, tasks, and collaboration, providing insight into their actions within this collaborative context.

The remainder of this paper is structured as follows. Section II provides a review of relevant literature on task allocation and adaptation within the context of Human-Robot Collaboration (HRC) and Human-Robot Interaction (HRI). Section III presents the problem statement and introduces the proposed framework. Section IV delves into the study's design, and the implementation of the planning and estimation method, and outlines the study procedure. In Section V, we initially summarized the results pertaining to human perception of the robot, collaboration, and tasks. Subsequently, we analyze the results, focusing on the robot's planning and its effectiveness in adapting to participants. Finally, Section VI concludes the paper, highlighting its limitations and proposing potential directions for future work.

II. Related Work

In this section, we initially delve into related research concerning task allocation and the incorporation of human preferences into task allocation and planning.

A. Task Allocation

Task allocation in HRC involves a suitable allocation of tasks to the human and robot agents and finding a proper chronological order for completing tasks based on problem-dependent decision factors and constraints. We can categorize task allocation approaches into two main groups: offline and online.

1) Offline Task Allocation: In offline task allocation, typically, the goal is to assign tasks based on prior knowledge of the suitability of agents for each task. As a measure of suitability, one can consider the proportionality of agents' abilities and constraints to the tasks' requirements and constraints. After determining suitability measures, the task allocation can be done by an expert [17], [18], via simulation studies [19], [20], or through mathematical modeling and optimization [12], [21]. In [19], after deciding possible suitable agents for each task, different cases are evaluated by simulation, and the one with the best utility value is selected. Similarly, in [20], multiple criteria are calculated through simulations, and a depth-first search algorithm is employed to find an optimal solution. In [12], the authors designed a human capability-based cost function to minimize human risk factors. Then, they applied their method in a user study, using the A^* algorithm for role assignment. In [21], the problem of disassembly sequence planning is formulated as an optimization problem to minimize the disassembly time while considering resource and safety constraints.

2) Online Task Allocation: Offline task allocation approaches force the human-robot team to adhere to the (optimal) plan obtained offline. However, in many real-world scenarios, uncertainties that might arise due to individual preferences and behaviors, alterations in the workspace, and changes in task requirements question the applicability of these approaches. Online task allocation methods aim to cope with this limitation by endowing the system with online replanning abilities. Typically, these methods fall into two groups. The first group of approaches, similar to the offline methods, finds an optimal task allocation, and the agents must follow the obtained plan. However, these methods are able to reactively replan online when the current task allocation is not valid anymore [10], [11], [22], [23].

The second group of methods mainly relies on the human agents' decisions, and the robot agents play more of a supportive role by adapting to the human agents. These methods do not create a fixed allocation requiring agents to adhere to it. In these methods, the human agents can make their own decisions and manage arising uncertainties, and the robot agents need to adapt their decisions proactively. In [24], the robot infers the human preference by a two-stage clustering approach and provides the parts for the human in an assembly task. Other work proposed a real-time decision-making mechanism for a cobot based on the human's short-term and long-term behaviors [25]. In [26], the robot infers human preferences and chooses one of the three operation modalities: 1- the robot plans, 2- the human plans, and 3- the robot adapts.

3) Human Preferences in Task Allocation: As discussed in the preceding paragraph, it is crucial for the robot to proactively infer human preferences when taking on a supportive role. However, incorporating human preferences in situations where the robot or a central unit (e.g., a manager, central computer, etc) is actively involved in task allocation and planning is an aspect that is often underestimated. This aspect, central to our research, has been demonstrated as a critical element in enhancing humans' positive perception of the robot [26]–[28]. In [27], different scenarios were explored in which the manager, the robot, or the participants themselves assign tasks. In [28], human participants' task type preferences are considered while balancing the assigned workloads. In [26], the system can adjust to user preferences, enabling the human partner to issue commands, or take a more passive role and seamlessly transition between different modes as needed. Nonetheless, most of these studies have considered human agents' preferences during offline planning, often by direct inquiry. What is notably missing is an assessment of participants' performance to determine whether their preferences and performance align with the team's overall performance or potentially hinder it.

B. Adaptation

As noted in the task allocation literature, robot adaptation is an integral ability for a robot to collaborate smoothly and efficiently with humans. This adaptation can be achieved through manually teaching the robot by experts (i.e., how to collaborate effectively while adhering to human preferences) [29], [30], equipping the robot with the ability to learn and adapt itself autonomously, or a combination of both. Manual instruction by experts, while fast and effective, faces practical limitations, including time consumption, challenges in conveying nuanced instructions, and scalability issues. [31].

Learning-based adaptation approaches typically employ supervised or unsupervised learning-based algorithms. The latter involves identifying decision factors (features) influencing human behavior and collecting behavioral data, which is then enriched through expert annotation or participant surveys [32]–[34]. The former, however, enables machines or robots to learn human preferences by observing their behavior and actions without the explicit need for annotation or data labeling [24], [35]–[37]. Some research also leverages both learning methods and experts' knowledge [31], [38].

However, these studies often emphasize the robot's adaptation to the human agent. In contrast, mutual adaptation — where both the human and robot adjust based on each other's actions and feedback — can be essential in human-robot collaboration. The authors of [39] explored mutual adaptation between humans and robots, where the robot initially guides the adaptable human but may unilaterally adapt if the human insists on their poor or inadequate performance or decisions.

III. Task Allocation & Planning

A. Problem Statement

This paper considers a collaborative task, τ, comprising two agents: a human and a robot. These agents must cooperate to complete a set of precedence-constrained subtasks, denoted as $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$. Each subtask τ_i has associated completion times, represented as t_i^h for the human and t_i^r for the robot. However, owing to uncertainties stemming from agents and the environment, the task completion time may deviate from the initially specified duration.

In each decision-making step, an agent has the agency to allocate a set of feasible subtasks to the other agent, as well as to assign a subtask to itself and execute it. What sets this problem apart from conventional task allocation and scheduling problems is that the agents here have the autonomy to choose their actions and subtask assignments rather than being assigned specific tasks with predetermined instructions regarding what and when they should execute them. Throughout this collaboration, the robot needs to:

- estimate the human agent's leading/following preference,
- monitor the impact of the human agent's actions on the overall team performance continuously,
- minimize the collaboration cost(e.g., completion time) while adapting to the human agent's preference and performance,
- detect and address human errors, if applicable.

B. Planning Architecture

The planning architecture is illustrated in Fig. 2.

Tasks & Environment: the overall system or task the human and robot collaborate on.

Human/Robot: These two blocks represent the input provided by the human and robot and applied to the system. It's important to note that this is an asynchronous decision-making process, where the human and robot agents act and make decisions independently and at different times.

State estimator: During the collaborative process, the robot evaluates the human's actions and infers their inclination towards leading or following. Furthermore, it is responsible for monitoring the human's performance and assessing their level of performance. These states, however, cannot be measured directly, and the robot needs to infer them through the history of the interaction. To do so, the state observer takes the history of the human's actions, the robot's beliefs and schedule, the human's internal states (e.g., speed and fatigue), and the tasks' states.

Robot Planner: The robot planner is responsible for providing the robot with a schedule based on the tasks and environment's states and the output of the state estimator block, belief about the human agent's preference and performance. The robot planner consists of two phases: task selection and task scheduling. In each decision step, when necessary, the robot performs task selection and subsequently performs task scheduling to determine its following action.

C. Planning Strategy

At each decision step, the robot planner must determine a one-to-one subtask assignment for the agents and establish a task execution schedule to minimize the collaboration cost, injecting both the human agent's preference and performance. Task allocation and scheduling problems can usually be modeled as mixed linear integer programs (MILP). However, the complexity of MILP-based solutions makes them computationally intractable. In addition, involving the robot's belief about the human agent's preference and performance adds more complexity to the problem due to the dynamic and unpredictable nature of human behavior and intentions. These challenges, in concert, make formulating and solving the problem as a single optimization problem increasingly demanding and arduous. Decomposing task allocation and scheduling is a promising and commonly used approach to deal with this complexity [28]. Here, we also take advantage of this idea and split the problem into two subproblems: task allocation and task scheduling.

In the first step, considering the agents' set $A = \{human, robot\}$, the task $\tau = \{\tau_1, \tau_2, \ldots, \tau_m\}$, and the subtasks' associated assignment costs to the human and robot $C_{\tau_i}(a), a \in A$, the robot first seeks an optimal task allocation. m is the number of subtasks that are not completed yet or need to be fixed. Subsequently, if necessary, a new set of subtasks $\tau_{new} = \tau \cup \tau_{allocate}$ is generated, including actions needed to allocate subtasks to the human $\tau_{allocate}$. For example, in a sorting task, the robot may place a box on the human agent's side to indicate that the sorting of this box has been assigned to them. This additional subtask, which requires a certain amount of time, is needed as part of the task allocation process. The task scheduler utilizes the derived optimal task allocation and τ_{new} to determine an optimal task schedule. If the solution achieved in the task allocation phase proves infeasible during the task scheduling phase, the first step must be repeated to obtain a revised allocation.

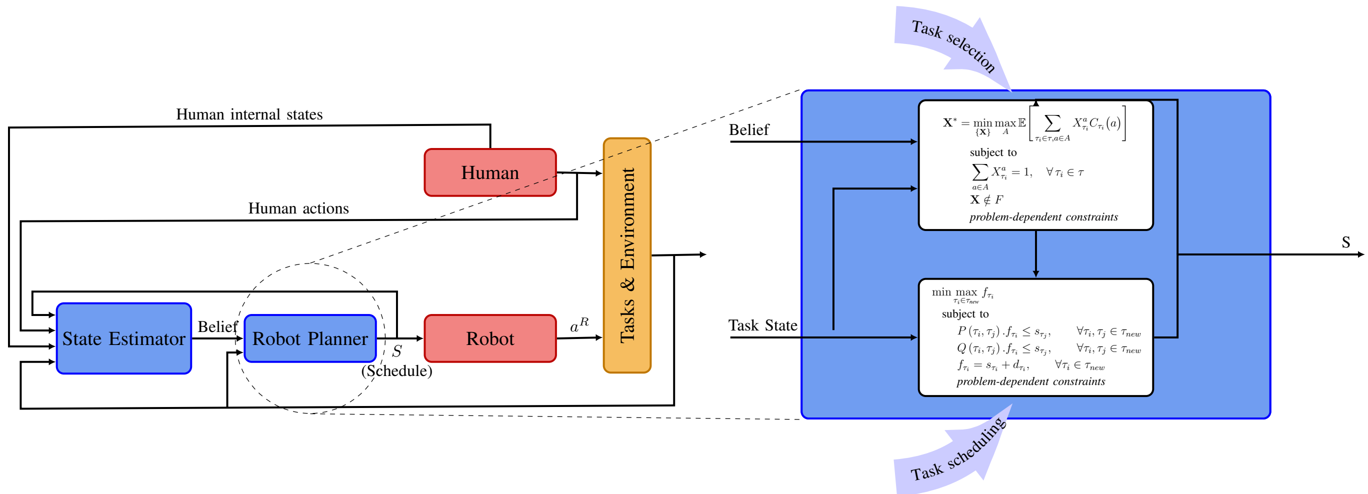


Fig. 2. Task selection and planning architecture encompassing the state estimator and robot planner. The robot planner takes into account human preference, performance, and task state, followed by the implementation of two planning steps: task selection and task scheduling.

The task allocation optimization problem can be formulated as (1), minimizing the maximum cost of task assignments, between the human and robot,

$$\mathbf{X}^* = \min_{\{\mathbf{X}\}} \max_{A} \mathbb{E}\left[\sum_{\tau_i \in \tau, a \in A} X^a_{\tau_i} C_{\tau_i}(a)\right] \tag{1}$$

subject to

$$\sum_{a \in A} X^a_{\tau_i} = 1, \quad \forall \tau_i \in \tau \tag{2}$$

$$\mathbf{X} \notin F \tag{3}$$

$$\textit{problem-dependent constraints}. \tag{4}$$

In Eq. (1), $\mathbf{X} = \left\{X^a_{\tau_i} \mid \tau_i \in \tau, a \in A\right\}$, where $X^a_{\tau_i} \in \left\{0, 1\right\}$ is a binary decision variable that indicates whether task τ_i is allocated to agent $a \in A$ ($X^a_{\tau_i} = 1$) or not ($X^a_{\tau_i} = 0$). Function $C_{\tau_i}(a)$ is the cost incurred by assigning task τ_i to the agent $a \in A = \left\{human, robot\right\}$ taking into account human agent's performance, and preference to follow the robot. Function C_{τ_i} imposes a higher cost for assigning the task to the human who prefers to lead, and a lower cost for allocating the task to a human with low performance, since having the robot allocate tasks limits the human's control and mitigates the potential for human errors. Equation (2) specifies each task must be assigned to exactly one agent, the human or the robot. Constraint 3 prohibits the task allocation solutions that do not lead to an infeasible task scheduling solution, F. Eq. (4) indicates that additional problem-dependent constraints can also be added to (1). After finding a solution for the task allocation problem, we generate τ_{new} with known required time, d_{τ_i}, to finish each task τ_i and updated task-precedence constraints.

In the subsequent phase, the robot determines an optimal schedule specifying both the tasks to be performed and their corresponding start and finish times. We introduce decision variables s_{τ_i}, representing the start times of subtask $\tau_i \in \tau_{new}$. f_{τ_i} denotes the finish time for subtask τ_i. To account for task precedence, we utilize a binary function $P(\tau_i, \tau_j)$, which takes the value of 1 when τ_i must be completed before τ_j. Additionally, we employ a binary decision variable $Q(\tau_i, \tau_j)$, where $Q(\tau_i, \tau_j) = 1$ indicates that both τ_i and τ_j are assigned to the same agent, and τ_i precedes τ_j.

In this paper, considering that we solely focus on collaboration time, the task scheduling problem can be formulated as minimizing the overall processing time:

$$\min \max_{\tau_i \in \tau_{new}} f_{\tau_i} \tag{5}$$

subject to

$$P(\tau_i, \tau_j) . f_{\tau_i} \leq s_{\tau_j}, \qquad \forall \tau_i, \tau_j \in \tau_{new} \tag{6}$$

$$Q(\tau_i, \tau_j) . f_{\tau_i} \leq s_{\tau_j}, \qquad \forall \tau_i, \tau_j \in \tau_{new} \tag{7}$$

$$f_{\tau_i} = s_{\tau_i} + d_{\tau_i}, \qquad \forall \tau_i \in \tau_{new} \tag{8}$$

$$\textit{problem-dependent constraints}. \tag{9}$$

Inequality (6) guarantees the precedence constraints. Inequality (7) maintains the constraint that each agent can handle only one subtask at a time. In (8), f_{τ_i} is determined by the time needed by the assigned agent, d_{τ_i}, to complete subtask τ_i. It's important to note that the constraints in this optimization problem are specific to the problem being addressed, and as such, the mentioned constraints can be adjusted, and new ones can be introduced (9).

Depending on the constraints in optimization problems (1) and (5), they can be formulated as mixed-integer linear or nonlinear programs. Nevertheless, in the collaborative scenario explained in the Appendix, we will formulate the problem as a mixed-integer linear program and subsequently use solvers such as Gurobi or CPLEX.

D. Algorithm

The task selection and planning procedure proposed in this paper is explained in Algorithm 1.

As indicated in Lines 1, the robot initializes its belief about the human agent's performance and following preference. Throughout the collaboration, until completing all subtasks, the robot monitors and records the human agent's actions and errors (Lines 3-4). The robot also has to update the task and

Algorithm 1: Task selection and planning

input : Precedence-constrained tasks, τ

1 $bel \leftarrow$ initial beliefs about human agent's following preference and performance

2 **while** *Tasks are not finished* **do**

3 Monitor the human's actions

4 Detect the human's errors

5 Update task, τ

6 Update bel

7 **if** *new schedule is needed* **then**

8 **while** *schedule, $S^\star$, is not found* **do**

9 $X^\star \leftarrow$ `TaskSelection`(τ, bel)

10 $\tau_{new} \leftarrow$ `CreatNewTask`$(\tau, X^\star)$

11 $S^\star \leftarrow$ `TaskSchedule`$(\tau_{new}, X^\star)$

12 $a^R \leftarrow$ `GetAction`$(S^\star)$

13 `ApplyAction`(a^R)

precedence constraints based on the finished subtasks and the subtasks need to be fixed (Line 5). Then, based on the planning structure shown in Fig. 2, the robot updates its belief about the human agent's following preference and performance (Line 6). Next, If the situation requires a new plan, as discussed before, the robot first solves for an optimal task allocation (Line 9) and then creates τ_{new} (Line 10). Next, the robot solves for an optimal schedule (Line 11). If the robot fails to find an optimal schedule with the current task allocation, it proceeds to generate a new task allocation. Subsequently, the robot performs its action, a^R based on the obtained schedule (Lines 12-13). This continues until the team finishes all subtasks.

The framework outlined in this section has been implemented on a mobile manipulator robot and evaluated in a user study, described below.

IV. User Study: Setup & Methodology

This section provides a detailed explanation of the collaborative scenario designed for the study and outlines the study procedure.

A. User Study Setup

The considerations for designing the user study scenario revolved around three aspects:1) **Collaboration:** focusing on the human and robot's planning ability, the cobot's better memory, and the human's faster speed, 2) **Leading/following preference:** the human's agency to adjust their leading/following role, 3) **Performance:** a task requiring cognitive load and a penalty for mistakes.

1) Setup: Fig. 3 and Fig 4 illustrate the experimental setup. The location of the camera in Fig. 3 shows the location where the picture in Fig. 4 was taken. The experimenter's table in Fig.3 also shows where the computer is located and where the experimenter stands. Both the human and the robot operate in designated work areas, distinctly demarcated by safety tape and cones to ensure clear separation. The collaborative task involves arranging colored blocks on four workspaces, namely

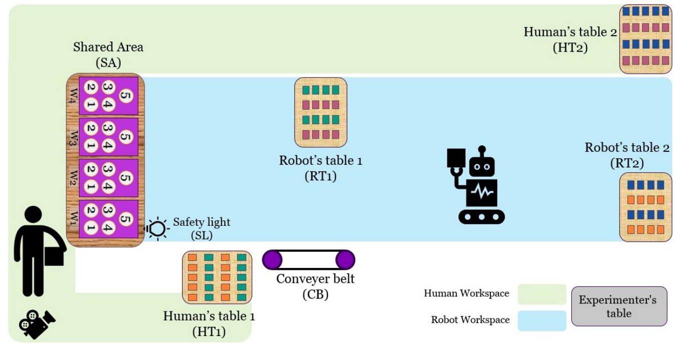


Fig. 3. The schematic of the experimental setup

Fig. 4. A view of the experiment environment, taken from the location of the camera in Fig. 3. The Fetch robot is positioned in its workspace between its two designated tables, as illustrated in Fig. 3. The nearest table in the figure, located next to the conveyor belt, belongs to the human and contains orange and green blocks. On this table, there is also a tablet on which the GUI is installed. The human agent's other table is situated at the corner of the room, containing blue and pink blocks. Four workspaces are present on the shared table, along with a light bulb. Safety tapes and cones separate the work areas of the agents.

W_1 to W_4, within a shared area (table). Each workspace comprises five numbered spots, and participants must adhere to the numerical order when placing blocks. For instance, in W_2, they must fill spot 1 before proceeding to spot 2. While the order of workspaces is flexible, allowing agents to switch between them, the adherence to spot numbering is paramount.

Within each agent's work area, two tables are present, each hosting different colors. In the robot's work area, the table with pink and green blocks is proximate to the shared table, while the one with blue and ornage blocks is situated at a distance. In the human agent's work area, a similar arrangement exists, with one table in close proximity containing orange and green blocks, and another distant table holding pink and blue blocks. Essentially, blue blocks are distant from both agents, while green blocks are close to both. Orange blocks are far from the robot but close to the human agent, while pink blocks are close to the robot and distant from the human agent. Table 1 provides a summary of the block distribution.

To complete the workspaces (spots), the team must adhere to a prescribed pattern of colors. The pattern shown in Fig. 5a represents a sample pattern the human-robot team is expected to follow when filling the spots. The structure of these patterns

TABLE I
DISTRIBUTION OF BLOCKS WITH RESPECT TO THE DISTANCE TO THE SHARED AREA

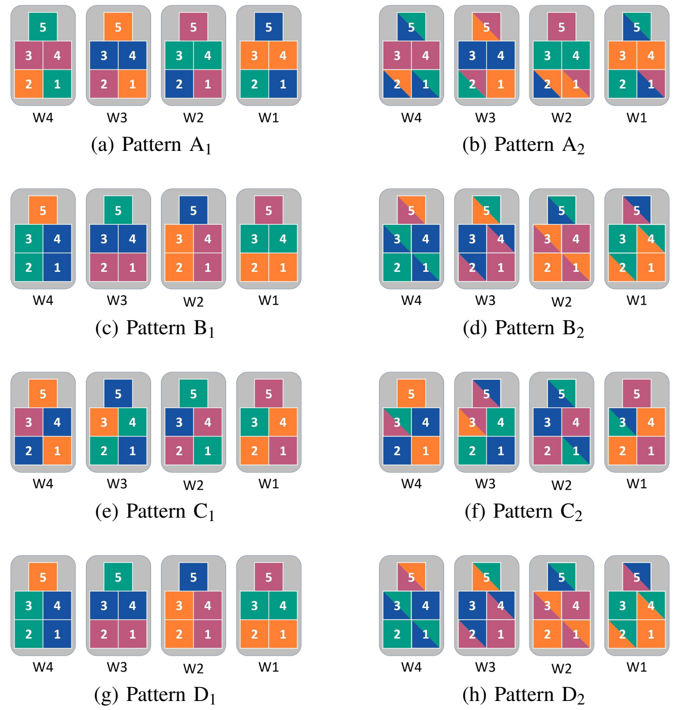
Color	Human	Robot
Green	Close	Close
Pink	Close	Far
Orange	Far	Close
Blue	Far	Far

precisely mirrors that of the workspaces within the shared area. Initially, participants are presented with a fully known version of the pattern (Patterns A_1, B_1, C_1, and D_1), such as A_1, printed on a sheet of paper. They are given 45 seconds to memorize it and then return it to the experimenter. Subsequently, the experimenter provides them with a partially known version of the same pattern (Patterns A_2, B_2, C_2, and D_2), for example, A_2. In this partially known version, certain spots contain two colors, with only one of them being correct. Essentially, the partially known pattern acts as a cue, aiding participants in recalling the initially presented pattern. Participants are allowed to retain the partially known version throughout the duration of the task.

As shown in Fig. 3, there is also a conveyor belt responsible for transferring the blocks that the robot needs to pass to the human agent. The robot needs to return the block to the human agent in cases where the human places a wrong block on the shared table and the robot decides to return it. Additionally, a red light bulb on the shared area warns the human agent not to place or pick up any blocks from the table as the robot is approaching. Participants, however, can continue planning, moving into their work area, and picking up blocks from other tables This, in addition to safety concerns, helps control the human agent's speed and prevents them from moving fast and perceiving the collaboration as a race.

For this study, we employed the Fetch mobile manipulator robot [40] and programmed it for autonomous navigation within its designated work area, performing the pick-and-place task. However, in consideration of safety, the experimenter maintains constant vigilance over the robot. If necessary, they can assume control using the joystick or promptly halt the robot's operations by pressing the emergency safety button. The following details provide additional information about the study setup:

- The collaborative scenario was inspired by the kitting task. The kitting task involves gathering a specific set of components for a defined purpose, which is then directed to workstations for the assembly of intermediate or end products [41].
- We chose the pattern memorization task to present participants with a cognitive challenge within the limitations of a brief collaboration scenario. Conducting a lengthy experiment that might mentally and physically strain participants would have been impractical and difficult to obtain ethics approval. Consequently, we selected relatively concise tasks, each lasting around 12-20 minutes, while ensuring they maintain mental engagement.



(a) Pattern A_1 (b) Pattern A_2 (c) Pattern B_1 (d) Pattern B_2 (e) Pattern C_1 (f) Pattern C_2 (g) Pattern D_1 (h) Pattern D_2

Fig. 5. **a, c, e, g:** Patterns A_1, B_1, C_1 and D_1 are the patterns, printed on the sheets of paper, that participants have to memorize in 45 seconds and then return it to the experimenter. **b, d, f, h**Patterns A_2, B_2, C_2 and D_2 are the patterns with partially unknown spot, and participants can keep it until the end of the collaborative task as a hint to recall the first pattern

- Patterns A_2, B_2, C_2, and D_2 feature 9, 12, 6, and 9 partially known spots, respectively. This intentional variation in the number of partially known spots aims to introduce distinct difficulty levels and cognitive load.
- Participants were instructed to handle only one block at a time, aligning with the cobot's gripper capacity to grasp a single block.
- The distribution of blocks includes ten of each color on the human's tables, while the robot's tables accommodate eight of each color. With eighteen blocks of each color in total, surpassing the required five per pattern, this distribution accounts for potential mistakes. The decision to place more blocks on the human's table acknowledges their faster working pace.
- Each block is equipped with an ArUco marker, facilitating the robot in locating and picking them up within the room.

2) *Tasks:* Each participant is asked to complete four tasks:

- Task 0: In this task, participants work alone, without the robot. For all participants, we use pattern A and follow the same procedure. This task, in addition to being a practice for participants to learn how to do the task and place the blocks, provides useful information regarding the participants' performance, self-confidence, and perceived workload, which its details and results are beyond the scope of the present manuscript and are detailed in [16].
- Tasks 1, 2, 3: The robot joins the human in these tasks.

TABLE II
THE HUMAN AND ROBOT'S SETS OF ACTIONS

Human	Robot
H1- Selecting a task for themselves	R1- Selecting a task for itself
H2- Assigning a task to the robot	R2- Assigning a task to the human
H3- Returning an block from the shared workspace	R3- Returning a *wrong* block from the shared workspace
H4- Performing a task assigned by the robot	R4- Performing a *correct* task assigned by the human
H5- Canceling a task assigned to the robot	R5- Canceling a task assigned to the human
H6- Rejecting a task assigned by the robot	R6- Rejecting a task assigned by the human

We consider six different permutations (modes) of the order of patterns B, C, and D (e.g., {B, C, D}, {C, B, D}) and randomly assign participants to one of these six modes in a way that has a balance across the modes. Following the same procedure, we asked participants to memorize the first pattern in 45 seconds and then return it. Next, we provide them with the second pattern. In addition, to resemble real cobot scenarios where mistakes have a cost associated, we informed participants that for each misplaced block on the table, at the end of the task, when they declare finishing the task, 1$ would be deducted from the total remuneration amount. Providing this disinformation, considered a type of deception, was approved by the University of Waterloo Human Research Ethics Board.

3) Agents' Actions: We considered a set of six different actions for each agent, the human and the robot. These actions, listed in Table II, are the same for both agents, providing them with a similar level of agency. Note, the feasibility of actions depends on the state of the task, and at each decision step, some of them may not be feasible. For example, when no task has been allocated to the robot, action H4, performing a task assigned by the human, is not applicable. Additionally, to provide the robot with greater autonomy in adapting its leadership role or reassuming it when human performance is poor, we intentionally designed the rejecting option (Action H6) to be less straightforward for the human (will be explained later). Action H6 involves the consecutive performance of H4 and H3 without actually performing them physically. Further elaboration on this will be provided in the subsequent paragraph. Fig. 6 displays the state graph of a single subtask, considering the possible actions of both human and robot agents as outlined in Table II.

4) Human-robot communication: Both agents need to communicate to inform each other about their next actions (see Table II). This can be done through a graphical user interface (GUI) designed and installed on a tablet. Participants could leave it on a table in the room or hold it. This helps them assign tasks to the robot and inform it about their next action. Similarly, the robot can assign tasks to the participant via the GUI and inform them about its actions. Fig. 7 shows a screenshot of the GUI. The GUI restricts the human agent from taking unfeasible actions, such as violating precedence constraints or choosing tasks already underway by the robot. We instructed participants how to work with the GUI and let them try it once before starting the tasks. Participants were also asked to scan the marker on the blocks before placing them on the shared area, as the robot has to know the block's ID if it needs to return it. This is implemented on the GUI, automatically launching the tablet's camera and letting participants scan the marker. We avoid the details of the GUI's design and implementation for the sake of brevity.

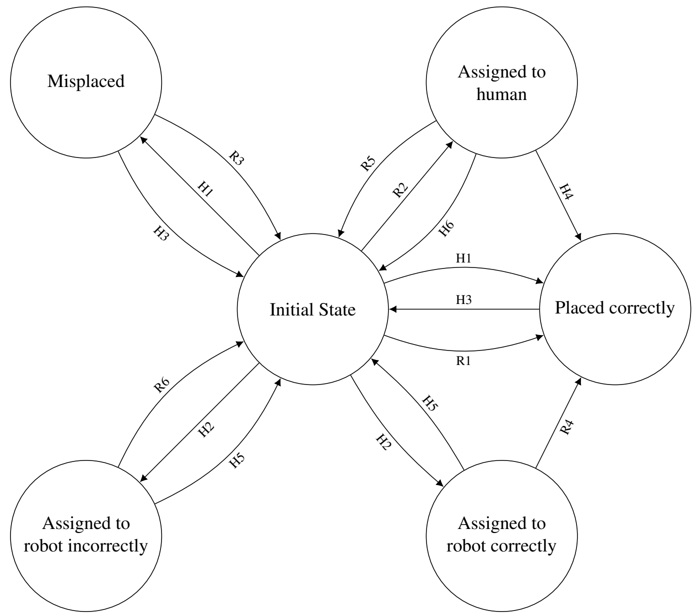


Fig. 6. State graph of a single subtask. There are six potential states for each subtask based on the actions detailed in Table 1. Initially, each subtask is in the "Initial state" state. To complete a subtask, it must transition to the "Placed correctly" state and remain in this state. The states "Assigned to robot correctly" and "Assigned to robot incorrectly" occur when the human agent respectively assigns a subtask to the robot with the correct or wrong color. The "Misplaced" state is reached when the human places a wrong color on the shared area for a subtask. The state "Assigned to human" happens when the robot assigns a subtask to the human agent.

Remark. If participants need to reject an assigned task by the robot, they first have to do the assigned task (H4) and then execute the returning action (H3), performing both actions on the GUI, without needing to do them physically.

B. Adaptation & Planning

Here, we briefly explain the components and some details of implementing the task planning architecture for the scenario designed.

Collaborative task: The tasks and their associated precedence constraints are conveyed using a directed acyclic graph termed the task graph. This graph depicts tasks as vertices,

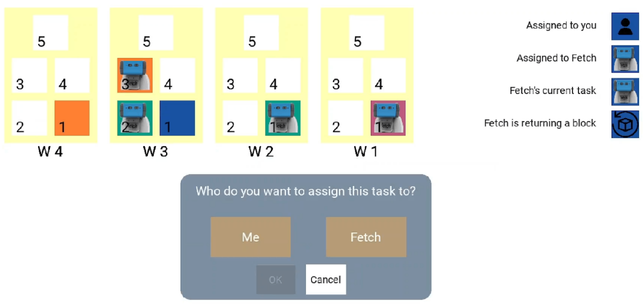


Fig. 7. A screenshot of the GUI through which participants can communicate their actions to the robot and receive information about the robot's decisions and actions (actions listed in Table II)

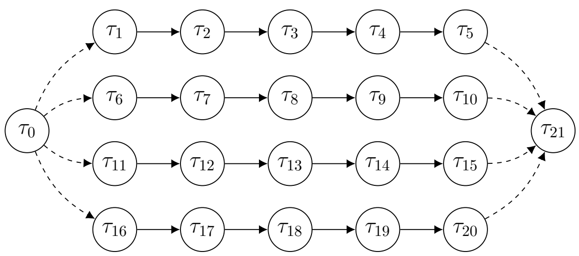


Fig. 8. Task graph of the experiment, including twenty subtasks (τ_1 - τ_{20}) and two dummy nodes representing starting (τ_0) and finishing points (τ_{21}).

while edges signify their precedence constraints. Fig. 8 provides a visual representation of the initial task graph for the experiment, including two dummy nodes designating the starting (T_0) and finishing (T_{21}) points.

In this experiment, we assume the same and time-invariant speed for all participants, the average human walking speed (1.3 m/s). In addition, we tested the required time for the robot to complete the pick-and-place scenario from different tables and used the average time for the near and far tables. Thus, the nominal required processing times for tasks are fixed for both the human and the robot.

Following preference and Performance: The robot uses the single scalar random variables α_f and α_e, respectively, to capture the human agent's following preference and performance (i.e., error-proneness or inaccuracy). For both, we consider a tuple of possible discrete values, $W = (w_0, \dots, w_{10}) = (0, 0.1, 0.2, \dots, 1)$ with steps of 0.1. In other words, there exists $i \in \{0, \dots, 10\}$ such that $w_i = \alpha_f$ (the same for α_e). For α_f, a value closer to zero indicates a leading preference, while values closer to one suggest a following preference. Additionally, values of α_e closer to one represent higher error-proneness (low accuracy), while values closer to zero indicate lower error-proneness (higher accuracy).

At the beginning of the session, as a default, the robot assumes that the human agent prefers to follow it and has high accuracy, so the robot sets its initial belief about the human agent as follows:

$$P[\alpha_f = w_i] = b(i; n = 10, p = 0.7),\ i \in \{0, \dots, 10\},$$

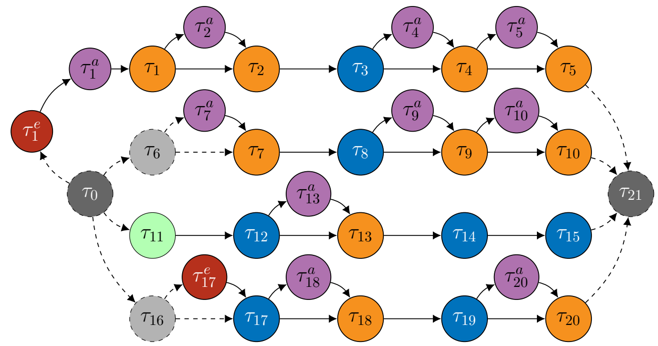


Fig. 9. Temporary task graph of the experiment after task allocation. Blue: Robot's tasks, Orange: Human's tasks, Cyan: Assigning tasks to the human, Red: Correcting human errors, Green: Already assigned tasks, Gray: Finished tasks

and similarly,

$$P[\alpha_e = w_i] = b(i; n = 10, p = 0.1),\ i \in \{0, \dots, 10\},$$

where $b(i; n, p)$ is a binomial distribution.

Task allocation: The problem of task allocation is formulated using (1), where

$$C_{\tau_i}(a) = \begin{cases} t_i^h \alpha_f + c_f(1 - \alpha_f) + c_v x_{\tau_i}^{robot} & a = Human \\ t_i^r + \alpha_e c_e + c_v x_{\tau_i}^{human} & a = Robot \end{cases}. \tag{10}$$

The parameter c_f represents the penalty incurred when assigning the subtask to a human agent who prefers to lead, while c_e denotes the penalty imposed for not allocating subtasks to the human agent who makes mistakes. Assigning subtasks to the human agent allows the robot to inform them about the next blocks to be placed on the shared table, minimizing the risk of wrong decisions by the human agent. Additionally, a penalty cost c_v is applied when the robot (human) is assigned a subtask that has already been allocated to the human (robot) agent, $x_{\tau_i}^{human} = 1$ ($x_{\tau_i}^{robot} = 1$). This penalty prevents frequent and substantial changes in task allocation. In addition, as a problem-dependent constraint, we need to add a constraint ensuring the allocation of at least one subtask from τ to the robot at each decision step. This constraint ensures the robot will start placing another task on the shared area or fix the human agent's errors. In Appendix-A, the task allocation problem is presented and reformulated as a mixed-integer program.

Fig. 9 illustrates an example of the updated task graph following task allocation. Subtasks 6 and 16 (τ_6 and τ_{16}) are shaded in gray to indicate their completion. However, for subtasks 1 and 17, the human agent mistakenly placed the wrong colored blocks on the shared area. Consequently, the robot must rectify these errors by executing subtasks τ_1^e and τ_{17}^e. Subtask 11 (τ_{11}) has already been assigned to the human. Additionally, the robot has allocated subtasks 1, 2, 4, 5, 7, 9, 10, 13, 18, and 20 to the human. Considering precedence constraints, the robot can initiate subtask τ_7^a to allocate and instruct the human agent through the GUI to perform subtask 7 (τ_7). The robot has also allocated subtasks 3, 8, 12, 14, 15, 17, and 19 to itself.

Task scheduling: After obtaining the optimal task allocation, the robot needs to find the optimal schedule by solving the optimization problem in 1. Appendix-B elaborates on the task scheduling optimization problem, representing it as a mixed-integer linear program.

Solving optimization problems: The task allocation and task scheduling problems, including their constraints, have been reformulated into mixed-integer linear programs, which are recognized as NP-hard optimization problems (see Appendix-A and Appendix-B). Our simulation experiments were conducted on a computer running Ubuntu 18.04, equipped with an Intel Core i7-11700 CPU with 8 cores operating at 2.5GHz and 16 GB of RAM. We utilized the GUROBI mathematical optimization solver, imposing a time limit to terminate the solver if it continues searching for additional solutions. We adopted a warm-start approach, providing the solver with a partially valid initial solution derived from the previous step's solution. Consequently, for the initial step, we could solve the problem offline.

Updating α_f and α_e: In updating the robot's estimate of the human agent's preference to follow and their performance, the robot needs to detect changes and adapt its planning accordingly. That is, the estimation method has to be sensitive enough to consider these changes. However, it also must not be oversensitive to affect the planning abruptly. For example, a single mistake in the human's decision making will not be reacted upon precipitously. Taking advantage of the "bounded memory adaptation model" approach, proposed in [39], we use a history of k-step in the past to estimate the human agent's preference and performance. A small value of k makes the estimation sensitive to changes, and a large value leads to measuring the overall leading/following preference and performance. In this work, we chose $k = 3$. Appendix-C provides details of the belief update and its required observation models, transition functions, and the human agent's action models, which we have designed to estimate the human agent's leading and following preference.

C. Recruitment

After obtaining ethics approval from the University of Waterloo Human Research Ethics Board, we initiated the participant recruitment process for the study by distributing recruitment flyers. This study involved three phases for each individual.

D. Study Procedure

1) Phase 0: In this phase, after potential participants reacted to our flyer and contacted the experimenter, we emailed participants the consent form and a set of questions regarding their familiarity with and prior experience in robots and artificial intelligence.

2) Phase 1 (In-Person): For this phase, we scheduled a 90-minute in-person session with each participant. In what follows, we explain the procedure.

Step 1: After greeting participants, we explained the setup and showed them their work area. We used some slides to explain all the details and inform them that:

- the robot may make mistakes in its decision-making (deception),
- the team would be penalized \$1 for each misplaced block at the end of the task, once participants confirm task completion (deception).

Step 2: (*Task 0*) We asked them to complete the task alone, without the robot, for Pattern A. Before starting the task, they answered a question about their self-confidence to accomplish the task, and after doing the task, they completed a questionnaire about the task load (NASA-TLX, [42]). Results from this step are not covered in this manuscript and are discussed in [16].

Step 3: We asked them to watch a video[1] of the Fetch robot performing pick-and-place, and then answer the questionnaire about their trust in the robot (Muir's questionnaire [43]).

Step 4: Participants worked with the GUI and practiced how to use it.

Step 5: (*Task 1*) Based on the mode assigned to participants (permutation of patterns B, C, D), experimenters gave them the associated pattern (as a sheet of paper) and asked them to memorize it within 45 seconds. Then, they returned the pattern (e.g., B_1) and were given the second pattern, a partially known version of the first pattern (e.g., B_2). Next, prior to starting the task, we asked them to answer two questions regarding their self-confidence and the expected helpfulness of the robot. Afterwards, they started the task and the collaboration with Fetch. Having completed the task, they answered three sets of questionnaires regarding their perceived task load, trust, and perception of the robot.

Remark. Participants start the task first, and the robot waits for them. They can allocate subtasks to the robot. The robot starts working as soon as participants allocate a task to themselves. This allows the robot to initially update its belief about their following preference.

Step 6: (*Task 2*) It followed the same procedure as *Task 1*

Step 7: (*Task 3*) It followed the same procedure as *Task 1*

Step 8: Finally, participants were asked to complete two sets of questionnaires. The first set focused on their performance as a team with the robot. The second set explored their collaborative experience using the short version of the User Experience Questionnaire (UEQ) [44], [45]. Additionally, participants were asked to rank the difficulty of tasks (Tasks 0-3) and respond to an open-ended question: *"Which abilities would you improve or add to Fetch if you were to use it in a manufacturing setting?"*.

The details and the results of the questionnaires on topics of participants' perception of the robot and collaboration (trust, helpfulness, task load, robot traits, team fluency, and user experience) go beyond the scope of this manuscript and are reported in [16]. However, we will use part of the results about participants' self-confidence, trust, and perceived workload for analyzing the proposed framework's performance.

3) Phase 2 (Online): For the online phase, we prepared a video of each participant's collaboration with the robot, only for Pattern B. This video contained the synchronized videos showing the room from two different angles, a screen

[1] https://youtu.be/ahZDo0_iyjg

recording of the GUI, and Patterns B_1 and B_2. The video created for one of the participants is available online[2]. During the online interview, we played the video, and participants were asked to talk about it and talk about their strategy, plan, and preference during their collaboration in the task in the video, as well as two other tasks (Patterns C and D). Then, we asked them to complete two questionnaires about their leadership and followership styles. However, the results and analysis of these two questionnaires are not in the scope of this manuscript and are reported in [16]. Finally, as per the approved ethics application, we explained about the *"Deception"* elements in the study and asked them to sign another consent form to let us use their data. Participants were remunerated a \$30 gift card as an appreciation of their participation.

V. Results & Discussion

We recruited 58 participants. However, we had to exclude data from 10 participants for various reasons, including a bug in the robot's program and the robot's failure. Consequently, our data analysis is based on the remaining 48 participants, consisting of 22 females, 24 males, and 2 selection "others", with an average age of 24.02 $\pm$ 3.93. The majority were University students (44), 3 were postdoctoral or visiting researchers, and 1 was a staff member. The results of this study can be analyzed from three key perspectives: 1- participants' perception of the tasks, the robot, and collaboration, 2- participants' preference and performance, 3- the robot's actions and performance. The first two perspectives were explored in [16]. Building on their findings, this paper focuses on the latter. Furthermore, we delve into specific participant cases to illustrate how the robot adapted to individuals and various situations.

Remark. We used the Kruskal–Wallis H test, a nonparametric statistical test, to determine whether there are statistically significant differences between two or more groups. When a significant overall difference exists among multiple groups, we employ the Dunn test as a post hoc analysis to identify specific group differences.

Remark. Analyzing the results based on the tasks corresponds to the chronological sequence, commencing from Task 0 and concluding with Task 3.

A. Highlights from Subjective & Objective Analysis

Here, we summarize the findings from analyzing 1- participants' perception of the tasks, the robot, and collaboration and 2- participants' preference and performance. These results are explored and elaborated in [16].

1) Subjective assessments reveal an improvement in participants' perception of the robot and collaboration following their interaction, along with a reduction in perceived workload.
2) Both subjective and objective assessments demonstrate that the robot effectively assisted participants in enhancing their performance and reducing errors.
3) The interview results show that most participants preferred to take on the leading role and have more control over the robot. Based on participants' preferences, we categorized them into four groups (from highest leading preference to highest following preference): 1- lead (17 participants), 2- collaborative-lead (20 participants), 3- collaborative-follow (4 participants), and 4- follow (3 participants).
4) According to the interviews, participants found the robot to be slower than themselves and preferred to handle more blocks.
5) The results indicate that participants, in general, found Pattern B more difficult compared to Patterns C and D. In addition, there was the highest number of participants who made at least one mistake in Pattern B. This can be attributed to the fact that Pattern B was both challenging to memorize and had the most unknown spots, leading participants to rely on the robot.

[2]https://youtu.be/X6Rj0zwQhz8

B. Robot's Actions and Estimation

We briefly highlighted participants' preferences and performance based on the interviews and recorded data from the user study. However, we must also explore how the robot could adapt to participants' preferences and performance.

1) Participants Preference: The robot updated its estimation of the human preference based on the 3-step history of the human's actions. The robot needs to consider the human agent's preference changes and adapt accordingly. However, we needed to create a measure to evaluate the robot's performance in estimating the human overall preference. To do so, first, we normalize the completion time and then fit a polynomial (e.g., degree 4) on the estimated values, $f(t), t \in [0, 1]$. Then, we calculate the area under the curve in a certain range, as a measure of participants' overall preference, $op = \int_{t_0}^{1} f(t)$, where we considered $t_0 = 0.2$.

To evaluate the robot's ability to estimate participants' preference, for each of them, we measured the average of the overall preference (*op*) in Tasks 1, 2, and 3. Combining them with the information gathered in interviews (i.e., participants' actual preferences) leads to Fig 10, which shows that the robot effectively estimated participants' actual preferences.

In addition, we analyzed participants' estimated overall preference for each task and noticed no significant difference among them ($H(2) = 2.29$, $p = 0.32$). This is justifiable as the majority of participants preferred to lead the robot.

2) Task Difficulty - Following Preference: As stated in the highlights of the participants' subjective and objective measurements, participants found Pattern B more challenging than Patterns C and D. Analyzing participants' estimated overall preference based on the patterns, as illustrated in Fig. 11, shows that there is a significant difference in participants' following preference based on the patterns ($H(2) = 6.5$, $p = 0.039$), indicating a significantly higher preference for following in Pattern B compared to Patterns C and D.

Based on the robot's task planning algorithm, the greater the human agent's preference for following or the occurrence of errors, the more tasks are assigned to them by the robot.

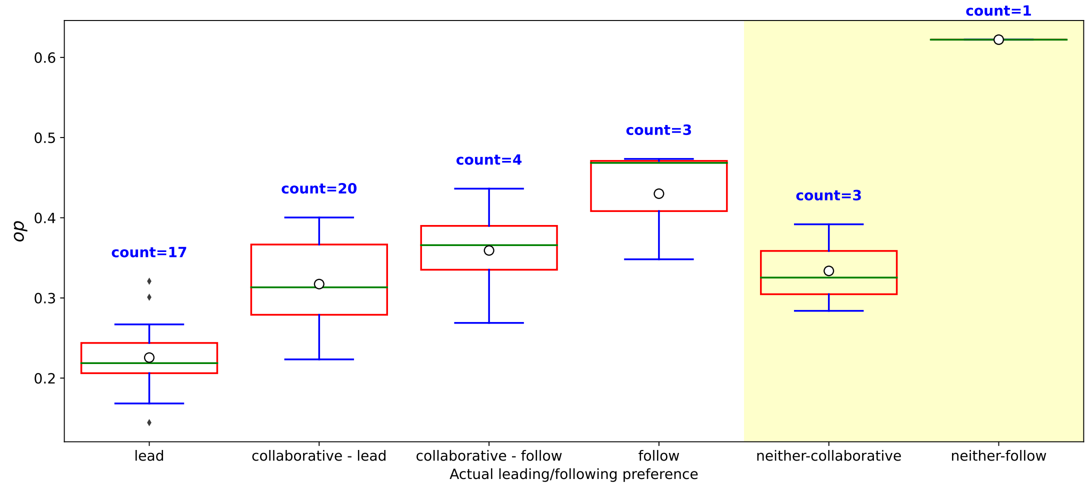


Fig. 10. The robot performance in estimating participants' preference by comparing with the participants' actual preference, gathered through interviews. The numbers above each bar show the number of participants falling into that group.

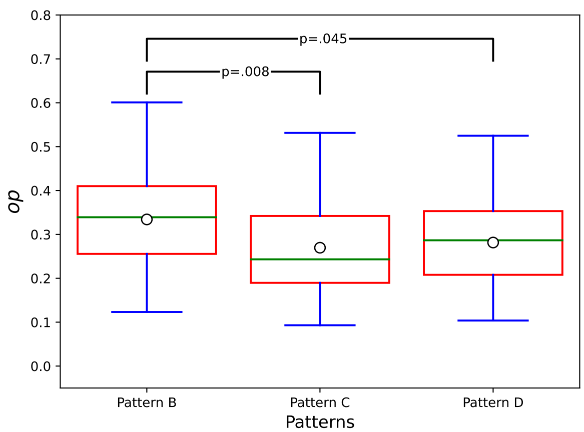


Fig. 11. Participants' overall estimated leading/following preference based on the patterns. The robot's estimation of participants' preference to follow was significantly higher in Pattern B compared to Patterns C and D.

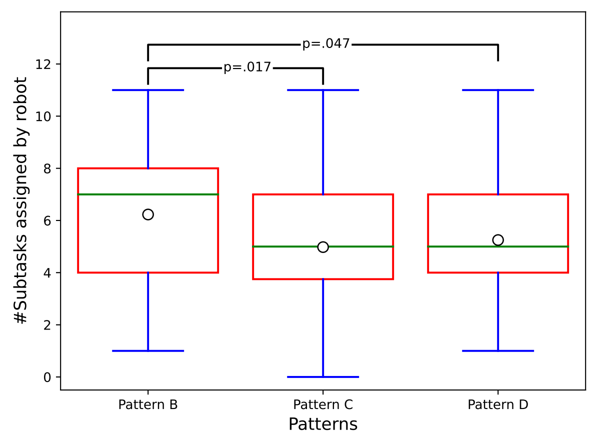


Fig. 12. Subasks assigned to participants by the robot based on the patterns. The number of assigned subtasks by the robot to participants was significantly higher in Pattern B compared to Patterns C and D.

Fig. 12 shows that the robot significantly assigned more subtasks to the human in Pattern B compared to Patterns C and D ($H(2) = 6.32$, $p = 0.042$). Likewise, this is justifiable as Pattern B was more difficult than the two others, and participants made more mistakes or preferred to follow the robot, and the robot could accordingly adapt its planning.

3) Task Assignment & Distribution: Referring back to the distribution of the blocks (Table I) and the lower speed of the robot compared to the human, the optimal task allocation will be closer to assigning pink blocks to the robot and blue and orange blocks to the human. Fig. 13 shows the colors of blocks completed by participants and those assigned by the robot. This distribution, resulting from the interplay between the robot and participants, is close to optimal. As expected, most of the orange blocks were completed by the human agent due to participants' expected rational decision-making and the robot's assignments. Similarly, pink tasks were completed by the robot. Regarding the blue subtasks, on average, participants completed most of them. However, the distribution of assigned and completed blue subtasks by the robot and human ranges from 0 to 5. This variation is due to some participants, as indicated by interviews, who preferred to assign more blue tasks to the robot, reducing their physical effort at the expense of longer collaboration time.

Following preference vs. Task distribution: Based on the designed algorithm, we expect the robot to assign more subtasks to a human with the following preference, which is particularly important for blue subtasks. Additionally, we are interested in examining how the robot's estimation of participants' following preference relates to the number of subtasks participants assign to it. As shown in Fig. 14, there is respectively a strong positive correlation and a strong negative correlation between the number of subtasks the robot assigned to participants and the number of subtasks that the human assigned to the robot with the estimated participants' following preference. This aligns with the designed algorithm.

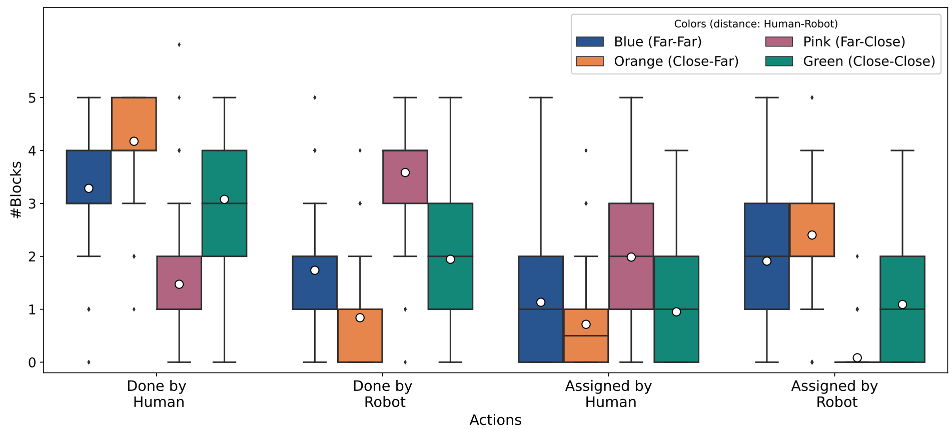


Fig. 13. Color of block that were done by human and the robot as well as were assigned by the robot or human to each other

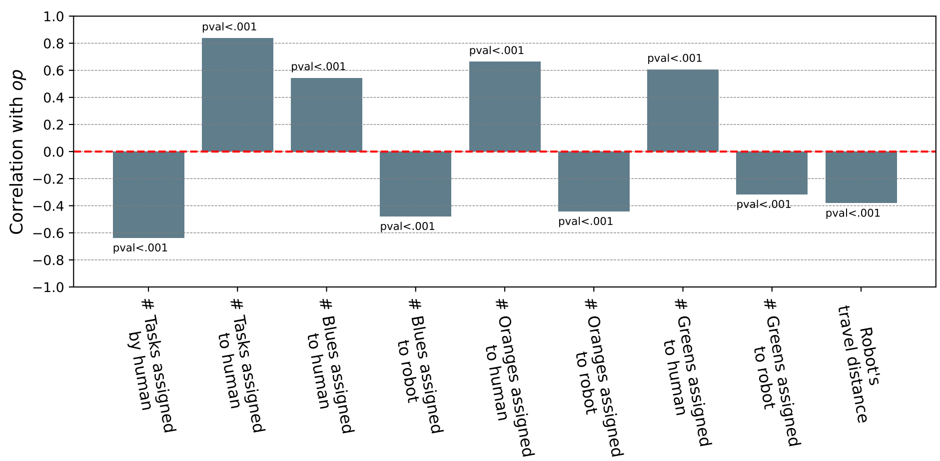


Fig. 14. Correlation between estimated participants' following preference and assigned/done tasks by the robot and participants

Furthermore, we can observe a moderate negative correlation between the number of assigned blue and orange tasks to the robot by humans and their following preference. Conversely, a moderate to strong positive correlation exists between the orange, green, and blue blocks allocated to the human agent and the estimated following preference. A weak negative correlation exists between the number of assigned green tasks and participants' following preference. As expected, the correlation between the approximate robot's travel distance and the estimated following preference shows a moderate negative correlation, as when the robot has a more leading role, it assigns tasks that are farther from itself to the human agent.

Discussion: The results indicate that the robot could successfully identify participants' following preferences and adapt its planning accordingly. We also observed that the robot's estimation of participants' following preference was higher for Pattern B than for Patterns C and D, resulting in more tasks being assigned to participants. This aligns with the previous finding that Pattern B was more challenging than the other two patterns, and participants required more support from the robot. All of these, in concert, showed that the robot was more of a leader in challenging tasks. This could result from either the participants' choice to follow the robot or the significant number of mistakes they made. Additionally, the robot could guide participants toward optimal task allocation to minimize collaboration time, taking into account the block's location and the fact that it is slower in comparison to the human agent.

C. Participant-Specific Analysis

We discussed the overall participants' preferences and performance. The main goal of the proposed framework, however, is to track changes in participants' performance and preferences. To evaluate the framework's online adaptation and planning ability, we discuss the results for some specific participants.

1) Leading or collaborating-leading Preference with a high accuracy: These participants preferred to lead the team and had a high accuracy. Thus, the robot estimated their leading preference and gave the leading role to them. Fig. 15a and 15b show the robot estimate of two participants' error-proneness and following preference, with respectively leading and collaborating-leading preferences.

2) Leading preference with occasional robot support: One of the participants preferred to lead the robot and minimize her physical effort by assigning most of the tasks to the robot. She also approved this in her interview and mentioned that she followed the same strategy in all three tasks. However, for Pattern B, she forgot the part of the pattern and thus let the robot assign her some tasks. She was also unsure about the last row of Patterns C. In Fig. 15c, the robot's estimation of her preference and performance is shown. Fig. 15d shows the robot estimations for another participant with a leading preference who was unsure about pattern B.

3) Leading preference with occasional errors: In this case, some participants took the leading role, and, in total, they could recall the pattern, but they made a few mistakes at some points. The robot detected the errors and updated its belief about their performance with usually a slight and temporary rise of α_e. Fig.15e, 15f, and 15g show the robot's estimate of three participants' preference and performance who had occasional errors. In Fig.15e, the participant made some consecutive mistakes at a point, but she improved her performance with the help of the robot.

4) Leading preference with Sudden Performance Drop: This participant led the robot in Pattern B, and her performance was good until almost the end of the task. However, she started making mistakes at the end as she had forgotten and was confused about the last row of the pattern. She initially insisted on her wrong decisions, although the robot rejected her assignment and fixed her mistakes. Subsequently, the robot updated its belief about her performance (Fig. 15h). While considering her as a person with a leading preference, it reassumed the leading role and assigned more tasks to her, guiding her to the correct pattern.

5) Following or collaborating-following Preference: Fig. 15i and 15j show the robot estimate of two participants' error-proneness and following preference, who preferred to follow the robot. Therefore, the robot took the leading role and assigned subtasks to them.

VI. Conclusion and future work

We investigated if and how proactive task planning and allocation can improve the efficiency of human-robot collaboration. The missing part in prior literature is overlooking either the human agents' leading/following preferences or the human agent's performance. This is what we focused on in this study: balancing the human agent's preference and performance while maintaining collaboration and the human perception of the robot at a high level.

Based on interviews with participants, we categorized them, based on their following/leading preference into four groups: "lead", "collaborative-lead", "collaborative-follow", and "follow", with the majority falling into the first two categories. This means that participants would prefer to take on more of a leading role and have more control over the collaboration. This finding can guide the design of collaborative scenarios and collaborative robots. Furthermore, we compared this result, showing actual participants' leading/following preferences, with the robot's estimation of their preference. This analysis showed that the robot successfully inferred their preference in most cases.

The results indicated that for more difficult tasks, participants trusted the robot more than their own abilities, which led them to take relatively more following roles. Our proposed task planning method properly inferred this need and provided more help to participants by taking on more leading roles. The robot could also identify when participants struggled to remember the correct patterns and made errors, and accordingly, it fixed their errors and provided more help.

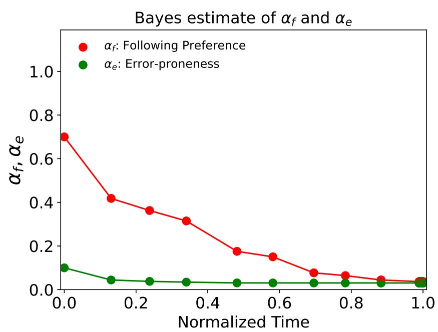
We also analyzed the distribution of subtasks between the robot and the human agent, showing that, overall, the interplay of participants and robot agents led to near-optimal task allocation. The results also indicated a moderate to strong correlation between estimated participants' preferences and measures of task distribution. Additionally, the results showed that participants were allocated more subtasks in tasks that were more difficult since they preferred to follow the robot or made many errors, causing the robot to take back the leading role.

In summary, we have developed a planning architecture for a robot, allowing it to adjust to its human teammate's preferences and performance while updating its plan in response to the task state and the human agent's actions. The findings indicate that most participants favored assuming a more leading role and exerting control over the team. The results also demonstrate the robot's capability to adapt its planning to provide assistance when required by its human teammate.

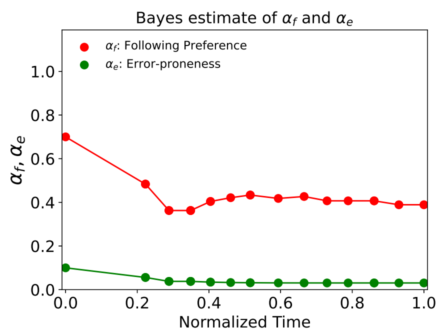
A. Limitations and Future Work

This work has certain limitations in terms of the study design and methods. The majority of our participants were young adults recruited from the University of Waterloo campus, while our ultimate target audience consists of working adults in settings such as manufacturing and warehouses. These two groups may have considerably different expectations and perceptions of a robot teammate. Recruiting actual working adults in those settings could help us create more practical collaborative settings and robots. Additionally, despite our attempts to simulate a working environment, e.g. by having an autonomous robot performing pick-and-place tasks and a setup including a conveyor belt, safety equipment, and a graphical user interface, future work could further improve the scenarios to make them as close as possible to a realistic environment, e.g. a warehouse automation or assembly environment. Another prevalent issue in manufacturing settings is the occurrence of sudden changes or unpredictable events that only humans could handle. Including these cases could also make the study closer to real-life situations.

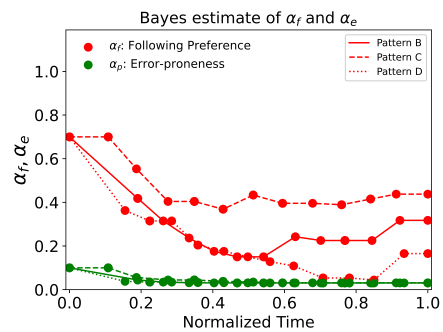
One of the interesting and important future research directions will be extending the framework to cases where there is a conflict between the agents, e.g. when both believe their decisions are correct. In this study, we assumed that the robot's decisions were always correct. However, in real-world settings, there can be cases in which the robot makes mistakes and may even not be aware of them, e.g., due to perception inaccuracies.



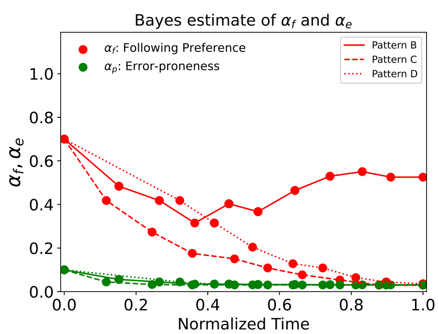
(a)



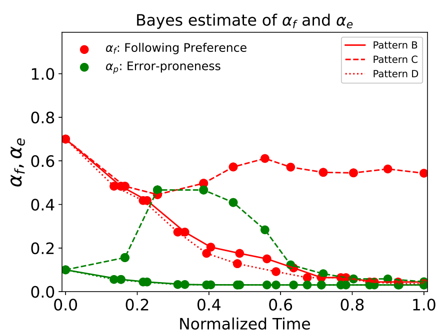
(b)



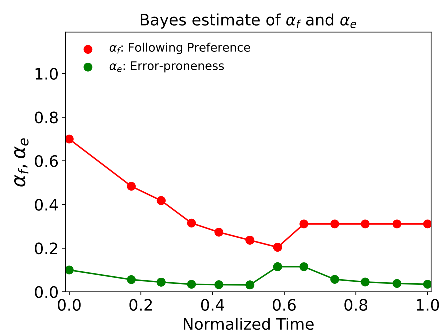
(c)



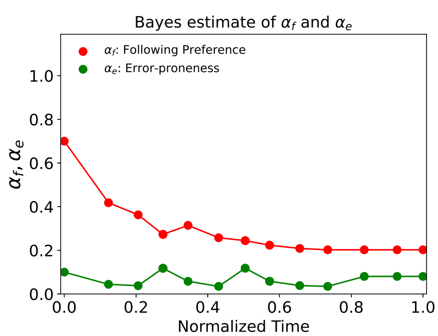
(d)



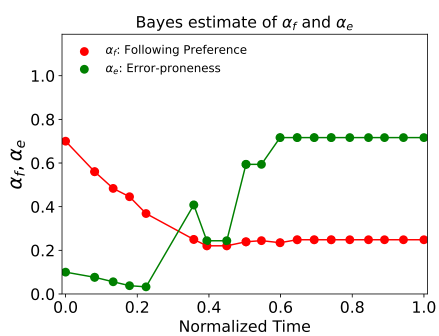
(e)



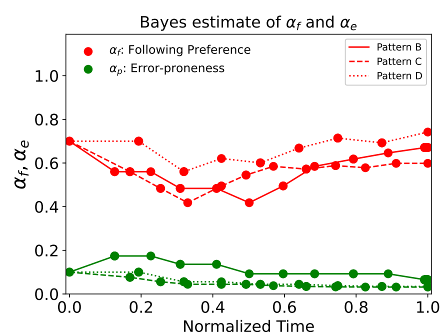
(f)



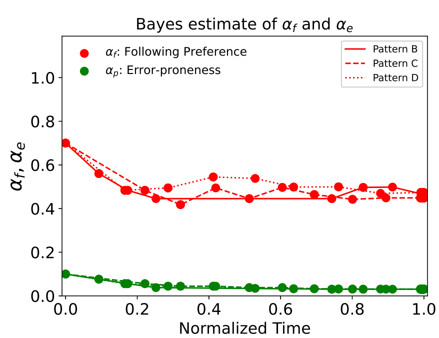
(g)



(h)



(i)



(j)

Fig. 15. The robot's estimates of the participants' preference and performance: a, b) Leading or collaborating-leading preference and high accuracy; c, d) leading preference with occasional robot support; e, f, g) leading preference with occasional errors; h) leading preference with sudden performance drop; i, j) following or collaborating-following preference

Furthermore, in this study, we considered the human agent's correctness in selecting block colors as a measure of their accuracy. This measure was easily understandable by our study participants, and we used it to assess their performance. However, practical measures can be added, such as completion time or travel distance. However, such metrics are not easily understandable and measurable for the human agents and could lead to conflict between the agents.

REFERENCES

[1] F. Vicentini, "Collaborative Robotics: A Survey," *Journal of Mechanical Design*, vol. 143, no. 4, p. 040802, 10 2020.

[2] S. El Zaatari, M. Marei, W. Li, and Z. Usman, "Cobot programming for collaborative industrial tasks: An overview," *Robotics and Autonomous Systems*, vol. 116, pp. 162–180, 2019.

[3] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, and N. Hagita, "Adapting robot behavior for human–robot interaction," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 911–916, 2008.

[4] H. Nemlekar, N. Dhanaraj, A. Guan, S. K. Gupta, and S. Nikolaidis, "Transfer learning of human preferences for proactive robot assistance in assembly tasks," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, p. 575–583.

[5] W. Wang, R. Li, Y. Chen, Z. M. Diekel, and Y. Jia, "Facilitating human–robot collaborative tasks by teaching-learning-collaboration from human demonstrations," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 2, pp. 640–653, 2019.

[6] E. C. Grigore, A. Roncone, O. Mangin, and B. Scassellati, "Preference-based assistance prediction for human-robot collaboration tasks," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4441–4448.

[7] A. Noormohammadi, A. Dahiya, A. M. Aroyo, S. L. Smith, and K. Dautenhahn, "The effect of robot decision making on human perception of a robot in a collaborative task - a remote study," in *Proceedings of the 9th International Conference on Human-Agent Interaction*, 2021, p. 423–427.

[8] A. Khamis, A. Hussein, and A. Elmogy, *Multi-robot Task Allocation: A Review of the State-of-the-Art*. Cham: Springer International Publishing, 2015, pp. 31–51.

[9] C. Schmidbauer, S. Zafari, B. Hader, and S. Schlund, "An empirical study on workers' preferences in human–robot task assignment in industrial assembly systems," *IEEE Transactions on Human-Machine Systems*, vol. 53, no. 2, pp. 293–302, 2023.

[10] A. Pupa, W. Van Dijk, C. Brekelmans, and C. Secchi, "A resilient and effective task scheduling approach for industrial human-robot collaboration," *Sensors*, vol. 22, no. 13, 2022.

[11] Y. Cheng, L. Sun, and M. Tomizuka, "Human-aware robot task planning based on a hierarchical task model," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1136–1143, 2021.

[12] E. Lamon, A. De Franco, L. Peternel, and A. Ajoudani, "A capability-aware role allocation approach to industrial assembly tasks," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3378–3385, 2019.

[13] K. Darvish, E. Simetti, F. Mastrogiovanni, and G. Casalino, "A hierarchical architecture for human–robot cooperation processes," *IEEE Transactions on Robotics*, vol. 37, no. 2, pp. 567–586, 2021.

[14] A. Noormohammadi-Asl, A. Ayub, S. L. Smith, and K. Dautenhahn, "Task selection and planning in human-robot collaborative processes: To be a leader or a follower?" in *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2022, pp. 1244–1251.

[15] ——, "Adapting to human preferences to lead or follow in human-robot collaboration: A system evaluation," in *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2023, pp. 1851–1858.

[16] A. Noormohammadi-Asl, K. Fan, S. L. Smith, and K. Dautenhahn, "Human leading or following preferences: Effects on human perception of the robot and the human-robot collaboration," 2024.

[17] C. Schmidbauer, S. Schlund, T. B. Ionescu, and B. Hader, "Adaptive task sharing in human-robot interaction in assembly," in *2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2020, pp. 546–550.

[18] R. Müller, M. Vette, and O. Mailahn, "Process-oriented task assignment for assembly processes with human-robot interaction," *Procedia CIRP*, vol. 44, pp. 210–215, 2016, 6th CIRP Conference on Assembly Technologies and Systems (CATS).

[19] P. Tsarouchi, G. Michalos, S. Makris, T. Athanasatos, K. Dimoulas, and G. Chryssolouris, "On a human–robot workplace design and task allocation system," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 12, pp. 1272–1279, 2017.

[20] G. Michalos, J. Spiliotopoulos, S. Makris, and G. Chryssolouris, "A method for planning human robot shared tasks," *CIRP Journal of Manufacturing Science and Technology*, vol. 22, pp. 76–90, 2018.

[21] M.-L. Lee, S. Behdad, X. Liang, and M. Zheng, "Task allocation and planning for product disassembly with human–robot collaboration," *Robotics and Computer-Integrated Manufacturing*, vol. 76, p. 102306, 2022.

[22] M. Faroni, A. Umbrico, M. Beschi, A. Orlandini, A. Cesta, and N. Pedrocchi, "Optimal task and motion planning and execution for multiagent systems in dynamic environments," *IEEE Transactions on Cybernetics*, pp. 1–12, 2023.

[23] S. Alirezazadeh and L. A. Alexandre, "Dynamic task scheduling for human-robot collaboration," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8699–8704, 2022.

[24] H. Nemlekar, J. Modi, S. K. Gupta, and S. Nikolaidis, "Two-stage clustering of human preferences for action prediction in assembly tasks," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3487–3494.

[25] O. C. Görür, B. Rosman, F. Sivrikaya, and S. Albayrak, "FABRIC: A framework for the design and evaluation of collaborative robots with extended human adaptation," *J. Hum.-Robot Interact.*, vol. 12, no. 3, may 2023.

[26] M. Fiore, A. Clodic, and R. Alami, *On Planning and Task Achievement Modalities for Human-Robot Collaboration*. Springer International Publishing, 2016, pp. 293–306.

[27] A. Tausch and A. Kluge, "The best task allocation process is to decide on one's own: effects of the allocation agent in human–robot interaction on perceived work characteristics and satisfaction," *Cognition, Technology & Work*, vol. 24, no. 1, pp. 39–55, 2022.

[28] M. Gombolay, A. Bair, C. Huang, and J. Shah, "Computational design of mixed-initiative human–robot teaming that considers human factors: situational awareness, workload, and workflow preferences," *The International Journal of Robotics Research*, vol. 36, no. 5-7, pp. 597–617, 2017.

[29] B. Akgun, M. Cakmak, K. Jiang, and A. L. Thomaz, "Keyframe-based learning from demonstration: Method and evaluation," *International Journal of Social Robotics*, vol. 4, pp. 343–355, 2012.

[30] J. Huang, D. Fox, and M. Cakmak, "Synthesizing robot manipulation programs from a single observed human demonstration," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 4585–4592.

[31] V. V. Unhelkar, "Effective information sharing for human-robot collaboration," Ph.D. dissertation, MIT, 2020.

[32] J. Ayoub, L. Avetisian, X. J. Yang, and F. Zhou, "Real-time trust prediction in conditionally automated driving using physiological measures," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–9, 2023.

[33] H. Soh, Y. Xie, M. Chen, and D. Hsu, "Multi-task trust transfer for human–robot interaction," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 233–249, 2020.

[34] C. Huang, W. Luo, and R. Liu, "Meta preference learning for fast user adaptation in human-supervisory multi-robot deployments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5851–5856.

[35] R. Luo, R. Hayne, and D. Berenson, "Unsupervised early prediction of human reaching for human–robot collaboration in shared workspaces," *Autonomous Robots*, vol. 42, pp. 631–648, 2018.

[36] S. Reddy, S. Levine, and A. Dragan, "First contact: Unsupervised human-machine co-adaptation via mutual information maximization," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 31 542–31 556.

[37] A. Kanazawa, J. Kinugawa, and K. Kosuge, "Adaptive motion planning for a collaborative robot based on prediction uncertainty to enhance human safety and work efficiency," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 817–832, 2019.

[38] A. Bajcsy, D. P. Losey, M. K. O'Malley, and A. D. Dragan, "Learning from physical human corrections, one feature at a time," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, p. 141–149.

[39] S. Nikolaidis, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in collaborative tasks: Models and experiments," *The International Journal of Robotics Research*, vol. 36, no. 5-7, pp. 618–634, 2017.

[40] M. Wise, M. Ferguson, D. King, E. Diehr, and D. Dymesich, "Fetch and freight: Standard platforms for service robot applications," in *Workshop on autonomous mobile service robots*, 2016, pp. 1–6.

[41] Y.-S. Tung, K. Bishop, B. Hayes, and A. Roncone, "Bilevel optimization for just-in-time robotic kitting and delivery via adaptive task segmentation and scheduling," in *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2022, pp. 524–531.

[42] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.

[43] B. M. Muir and N. Moray, "Trust in automation. part ii. experimental studies of trust and human intervention in a process control simulation," *Ergonomics*, vol. 39, no. 3, pp. 429–460, 1996.

[44] B. Laugwitz, T. Held, and M. Schrepp, "Construction and evaluation of a user experience questionnaire," in *HCI and Usability for Education and Work*, A. Holzinger, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 63–76.

[45] M. Schrepp, J. Thomaschewski, and A. Hinderks, "Design and evaluation of a short version of the user experience questionnaire (ueq-s)," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 6, pp. 103–108, 12/2017 2017.

APPENDIX

A. Task Allocation

The derivation of optimization problem 1 for the task within the designed scenario is equivalent to the optimization problem represented by 11, which is formulated as a Mixed-Integer Linear Program (MILP). In this context, we introduce decision variables $q_i \in Q$ for $i \in I = \{1, \ldots, n_t\}$ to determine the assignment of agents to subtasks, where n_t represents the total number of remaining tasks. Specifically, when $q_i = 1$, it signifies that subtask i is assigned to the human agent, while $q_i = 0$ indicates the allocation of the subtask to the robot.

$$Q^* = \arg\min_{\{q_i\}} z \tag{11}$$

subject to

$$z - \sum_{i \in I} q_i \Big(t_i^h p_f + c_f(1 - p_f) + c_h x_{\tau_i}^{robot}\Big) \geq 0 \tag{12}$$

$$z - \sum_{i \in I} (1 - q_i)\Big(t_i^r + p_e c_e + c_r x_{\tau_i}^{human}\Big) \geq 0 \tag{13}$$

$$\sum_{i \in U} q_i \leq |U| - 1, \tag{14}$$

where U is the set of indexes of all feasible subtasks that the robot can perform immediately by placing the blocks on the shared area. We ignore details of obtaining U. This inequality ensures that the robot will start placing a new block after placing the previous one.

B. Task Scheduling

After allocating tasks to both the robot, denoted as τ_{robot}, and the human, denoted as τ_{human}, such that $\tau_{new} = \tau_{robot} \cup \tau_{human}$, the robot proceeds to solve the optimization problem 15 in order to find an optimal task schedule. Within this problem, the decision variables $S = \{s_{\tau_i}\}$ dictate the start times of the subtasks. Binary decision variables $O = \{o_{i,j}\}$ are employed to determine if subtask τ_i comes before or after τ_j. The set V comprises indices for all feasible subtasks in τ_{robot} that the robot can immediately perform by placing the blocks on the shared area. Moreover, binary decision variables $B = \{b_{\tau_i}\}$ are used to indicate whether $\tau_i \in V$ begins at $s_{\tau_i} = 0$.

$$\min_{\{S,O,B\}} z \tag{15}$$

subject to

$$P(\tau_i, \tau_j) \,.\, f_{\tau_i} \leq s_{\tau_j}, \qquad \forall \tau_i, \tau_j \in \tau_{new} \tag{16}$$

$$s_{\tau_i} - f_{\tau_j} + M(1 - o_{i,j}) \geq 0 \qquad \forall \tau_i, \tau_j \in \tau_{human} \tag{17}$$

$$s_{\tau_j} - f_{\tau_i} + M o_{i,j} \geq 0 \qquad \forall \tau_i, \tau_j \in \tau_{human} \tag{18}$$

$$s_{\tau_i} - f_{\tau_j} + M(1 - o_{i,j}) \geq 0 \qquad \forall \tau_i, \tau_j \in \tau_{robot} \tag{19}$$

$$s_{\tau_j} - f_{\tau_i} + M o_{i,j} \geq 0 \qquad \forall \tau_i, \tau_j \in \tau_{robot} \tag{20}$$

$$s_{\tau_i} - M b_{\tau_i} \leq 0 \qquad \forall i \in V \tag{21}$$

$$\sum_{i \in V} b_{\tau_i} \leq |V| - 1 \tag{22}$$

$$z - f_{\tau_i} \geq 0 \qquad \forall \tau_i \in \tau_{new} \tag{23}$$

where M is a large positive constant. In this problem, we assume that rejecting the human agent's assignments and allocating tasks to the human takes zero seconds.

C. Updating α_f and α_e

To update the belief, we employ the belief update method employed by [39] to estimate human adaptability. According to this method, the system needs to be considered as a factorization of observable (X) and unobservable (Y) state variables of the system $S : X \times Y$. Subsequently, belief update can be computed as:

$$\begin{aligned} b'(y') =& \eta Z(x', y', a^R, o) \sum_{y \in Y} T_x\Big(x, y, a^R, a^H\Big) \\ & T_y\big(x, y, a^R, a^H, x', y'\big) \pi^H\big(x, y, a^H\big) b\big(y\big), \end{aligned} \tag{24}$$

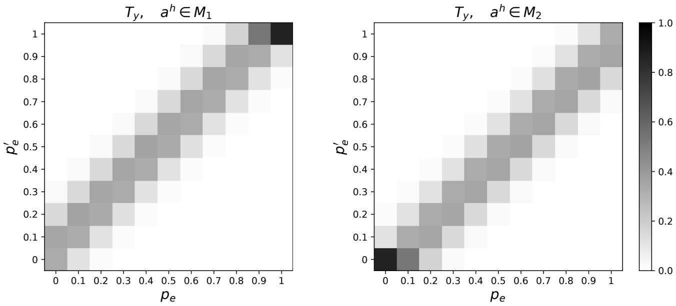
where T_x and T_y are the transition functions, z is the observation function, and π^H is the human action model (policy).

Following preference: We consider $Z = 1$ and $T_x = 1$. We also consider $T_y\big(x, y, a^R, a^H, x', y'\big) = \mathbb{I}(y = y')$, where $\mathbb{I}$ is an indicator function. This is based on the reasonable assumption that the human agent's preference changes infrequently and is usually fixed. The human agent's strategy is determined based on the analysis of the preceding three steps in their actions. The actions taken into consideration by the robot for updating its policy regarding human following preferences (P_f) include the actions of assigning a subtask to the robot (F_1), performing a subtask assigned by the robot (F_2), or refraining from performing a subtask assigned by the robot (F_3). By denoting the occurrences of F_1, F_2, and F_3 as f_1, f_2, and f_3, respectively, within the sequence of human actions spanning a history of three steps, the resulting human policy can be defined as follows:

$$\pi_f^H(x, y, a^H) = \begin{cases} \frac{\alpha f_1 + f_2}{\alpha f_1 + f_2 + f_3} y & a^H \in F_1 \cup F_2 \\ \frac{f_3}{\alpha f_1 + f_2 + f_3} y & a^H \in F_3 \end{cases}, \tag{25}$$

where $\alpha > 1$ is a parameter that weighs cases where the human assigns a subtask to the robot more heavily.

Human error: We consider $Z = 1$ and $T_x = 1$. Modeling human error, and specifically, the humans' memory model in this scenario, is demanding and not the focus of this paper.



(a) Wrong action

(b) Correct action

Fig. 16. Estimating the human's accuracy: Transition probability, T_y

However, we consider a simple model for T_y and π^H as they are required to estimate p_e. Defining $g_l(y)$ and $g_u(y)$ as the functions which return respectively the closest value less and closest value greater than y in set Y, we have

Modeling human error, specifically the human memory model in this particular scenario, presents significant challenges and falls outside the primary focus of this paper. Nevertheless, we adopt a simplified model for T_y and π^H as necessary components for estimating p_e. We define two functions, $g_l(y)$ and $g_u(y)$, which respectively identify the closest value less than and greater than y within the set Y. Consequently, we formulate T_y as follows:

$$T_y = \begin{cases} p\big(y' \leq Z < g_u(y')\big), & \\ \qquad Z \sim \mathcal{SN}\big(g_u(y),\, \sigma^2, \beta_1\big) & \text{if } a^H \in M_1 \\ p\big(g_l(y') \leq Z < y'\big), & \\ \qquad Z \sim \mathcal{SN}\big(g_l(y),\, \sigma^2, \beta_2\big) & \text{if } a^H \in M_2 \end{cases}, \tag{26}$$

where β is the skewness factor of the skew-normal distribution function $\mathcal{SN}(g_l(y),\, \sigma^2, \beta)$. The erroneous actions (M_1) and correct ones (M_2), if they are not assigned to the human by the robot, are taken into account for updating α_p. The transition probability T_y heatmap is illustrated in Fig. 16. Considering m_1 and m_2 as the frequency counts of M_1, M_2, the human error model is as follows:

$$\pi_e^H(x, y, a^H) = \begin{cases} \frac{m_2}{m_1+m_2} y & a^H \in M_2 \\ \frac{m_1}{m_1+m_2} y & a^H \in M_1 \end{cases}. \tag{27}$$