Enabling Embodied Human-Robot Co-Learning: Requirements, Method, and Test With Handover Task

Emma M. van Zoelen, Hugo Veldman-Loopik, Karel van den Bosch, Mark Neerincx, David A. Abbink, *Senior Member, IEEE*, and Luka Peternel, *Member, IEEE*

***Abstract*—Despite a large body of research on robot learning, it has not yet been thoroughly studied how collaborating humans and robots learn reciprocally. In such situations, both humans and robots continuously learn about each other and the task through interaction. This letter addresses the research question: *"How can human-robot co-learning be facilitated in physically embodied collaborative tasks?"*. First, we derived five requirements for successful human-robot co-learning from literature: shared goal, synchrony, interdependence, adaptability, and transparency. Based on these requirements, we designed a collaborative human-robot handover task and a robot Q-learning method. In an evaluation with six human participants co-learning was indeed found to emerge in the hand-over task. Particularly, for three of the human-robot dyads, our designed setup proved to facilitate co-learning in a way that met all five requirements. The task and robot learning method presented in this letter demonstrate how human-robot co-learning can be enabled in physically embodied tasks.**

***Index Terms*—Human-robot collaboration, physical human-robot interaction, reinforcement learning.**

I. Introduction

HUMAN-ROBOT collaboration research has rapidly evolved in the last decade [1]. Collaborative robots are being used in various industries and domains, performing an increasing amount of tasks side by side with humans. To ensure that humans and robots collaborate effectively, it is essential that they learn about each other and the task, to improve their collaboration over time [2]. Especially in dynamic, real-world environments, this learning will partly need to take place on the job, while humans and robots are collaborating.

Received 23 July 2024; accepted 28 November 2024. Date of publication 18 December 2024; date of current version 3 January 2025. This article was recommended for publication by Associate Editor Y. Hu and Editor G. Venture upon evaluation of the reviewers' comments. *(Corresponding author: Emma M. van Zoelen.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Human Research Ethics Committee at Delft University of Technology.

Emma M. van Zoelen and Mark Neerincx are with the Department of Intelligent Systems, Delft University of Technology, 2628, CD Delft, The Netherlands, and also with TNO, 2595, DA Hague, Netherlands (e-mail: E.M.vanZoelen@tudelft.nl; M.A.Neerincx@tudelft.nl).

Hugo Veldman-Loopik, David A. Abbink, and Luka Peternel are with the Department of Cognitive Robotics, University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: Hugo@Loopik.nl; D.A.Abbink@tudelft.nl; L.Peternel@tudelft.nl).

Karel van den Bosch is with TNO, 2595, DA Hague, The Netherlands (e-mail: karel.vandenbosch@tno.nl).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2024.3519875, provided by the authors.

Digital Object Identifier 10.1109/LRA.2024.3519875

We call this continuous collaborative learning process Co-Learning. When a human and robot are Co-Learning, both agents simultaneously learn how to collaborate effectively as a team by adapting their behavior to the other [3], [4], [5]. As a result of this reciprocal adaptation, new patterns of collaboration emerge. Human-robot Co-Learning can be used to improve performance and personalize robot behavior to their human collaborator [6]. Recent research has explored the use of Co-Learning to improve Collaboration Fluency and task performance [2], [3], [4], [7].

Co-Learning is a relatively new and unstudied topic within human-robot collaboration. The dynamics of Co-Learning have previously been explored in virtual environments [4] and with Wizard-of-Oz setups [3]. These initial studies have shown that humans and robots collaborating can develop successful patterns of collaboration as a result of reciprocal adaptations. There is however limited research on Co-Learning in physically embodied environments with robots whose actions are governed by Machine Learning algorithms. There are some relevant studies on human-robot mutual adaptation and collaborative learning (e.g. [8], [9], [10], [11]), within which there is often a strong focus on task performance improvement. There are only a few exploratory studies on the process of Co-Learning and the patterns of collaboration that emerge as a result [6], [12]. As the existing studies pay little attention to how Co-Learning can be facilitated, our research was guided by the following research question: *"How can human-robot co-learning be facilitated in physically embodied collaborative tasks?"*

We studied this question for a team consisting of a human-robot dyad, and focused on Reinforcement Learning (RL) as the learning method for the robot. This letter provides a set of core requirements for human-robot co-learning, and presents the design and evaluation of a human and a robot collaborating on a handover task based on these requirements. We use a qualitative approach to get an in-depth understanding of the co-learning process in relation to our requirements and design, as well as to provide a basis for future research on co-learning.

II. Design Requirements

We have defined five design requirements for human-robot Co-Learning in physically embodied tasks based on literature research [13]: shared goal, synchrony, interdependence, adaptability, and transparency. We examine each of them in the following subsections.

A. Shared Goal

Ensuring that both team members have the same goal is crucial for them to converge to congruent strategies [3], [7], [14]. This can be done by rewarding both team members based on the joint task goal (e.g., [15]). This leads to the first requirement:

R1: Both the human and the robot are rewarded similarly, based on their collaborative performance.

B. Synchrony

Co-Learning is most likely to succeed when both agents learn synchronously to enable continuity, reciprocity and complementarity in their learning process (cf., [16], [17]). If team members' learning is "disconnected", the motivation for collaboration can be lost due to uncertainty about the other's progress and contribution. One of the collaborators being ahead in learning could also cause a hierarchy in the team that could be harmful for interdependence [14]. Our second design requirement therefore states:

R2: The robot has the ability to learn in synchrony with the human team member.

C. Interdependence

To enable Co-Learning, all team members should be able to meaningfully contribute to the task by complementing and supporting each other. Such a team relationship is described by *interdependence*, which is a requirement for collaboration [7], [14], [18] and therefore for Co-Learning [2], [3], [4], [19]. Interdependence is often used to describe team and task designs in studies on team collaboration [14], collaborative performance [14], [19], team task design [4], [7] and team learning [2], [3].

Interdependence is built up of two types of dependence: 1) *hard dependence*, in which team members can only complete a task together, and 2) *soft dependence*, when team members do not strictly need each other to achieve the group goal, but have opportunities to collaborate to perform better as a team. This can lead to team members proactively adapting to and supporting each other, which is a vital part of the Co-Learning process. Soft dependencies have a recursive nature; when interdependence is established, soft dependencies can arise, retaining and strengthening the interdependent relationship. To enable Co-Learning, we therefore need to facilitate the formation of an interdependent relationship between the human and the robot. This is done by ensuring hard dependencies between the human and the robot and by creating opportunities for soft dependencies to emerge. The third requirement is as follows:

R3: The task design ensures hard dependencies and allows for soft dependencies between the human and the robot, in both directions.

D. Adaptability

In Co-Learning the robot algorithm must remain adaptable to change, because the human team member is learning at the same time and might therefore change its behavior later on [2]. This can cause certain state-action pairs, that were previously discarded by the robot algorithm, to now be preferred due to changes in the policy of the human. We therefore defined a requirement that ensures that the robot always keeps exploring:

R4: The RL algorithm can continuously adapt its behavior during all stages of the learning process.

E. Transparency

Mutual transparency is crucial for the understanding of each others' contribution to the joint task performance [20], [21], [22]. Team members' behaviors and decisions must be observable, predictable and directable in a collaboration [2], [7]. Both team members should be able to observe the state and actions of the other team member, to allow them to adapt to each other to develop patterns of collaboration. Mutual transparency helps to avoid hierarchical inequalities within the team and ensures that both team members are able to properly adapt. This leads to the fifth requirement:

R5: The human and the robot are able to observe and understand each other's state and actions.

III. Methods

A. Task Design

We designed a human-robot handover task, a common task found in physical human-robot collaboration [23]. To coordinate the specific moment in which the responsibility of not dropping the handover object switches from one team member to the other, the team members must collaborate to successfully complete the task. This ensures that a symmetrical hard dependency is embedded in the task. Moreover, passing an object involves multiple elements in which soft dependencies can arise. For instance, the position and orientation at which the object is handed over needs to be predicted or learned, thereby allowing for proactively reciprocating the strategy of the other team member. It also allows for different strategies (e.g. the robot drops the object while the human holds its hand up, or the robot conveys the object close to the human until the human seizes it). Therefore, there is space for a human-robot team to explore and learn what works well for their team. The presence of both hard and soft dependencies follows **R3**.

Additionally, the task of handing over an object is relatively short and can either succeed or fail. It is ideal for rewarding the team based on their collaborative performance (**R1**), and, as it is a short task, the team can rehearse the task often in a short amount of time. Therefore, the robot is rewarded on a regular basis, allowing it to continuously update its policy. This contributes to requirement **R2**, as it allows the robot to learn at a human timescale.

To accommodate **R3** more strongly, the task was designed such that responsibilities are divided over both agents, creating dependencies between the human and the robot. We describe the capabilities of the robot, defined by the State-Action space of the RL algorithm, in Section III-A1. We explain how we

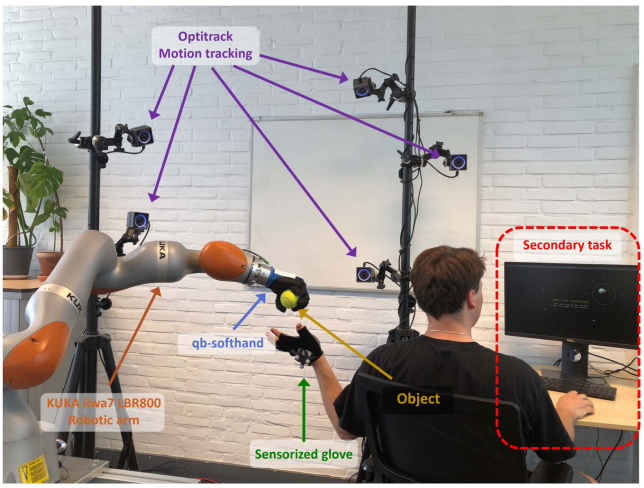

Fig. 1. Experiment setup for the human-robot co-learning of an object handover task. The robot consists of the *KUKA LBR iiwa7 800* robotic arm with the *qb-softhand* attached. The *Optitrack* motion tracking system is used to track the pose of the human hand via a sensorized glove. The human is also performing a secondary task, as explained in Section III-A2.

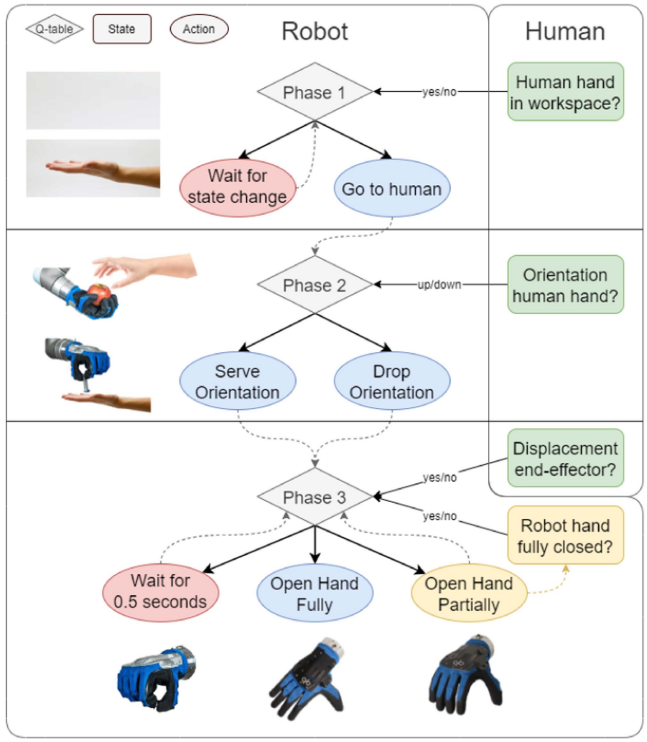

Fig. 2. A flow diagram that shows the capabilities of the robot throughout three phases of the task. The figure shows the binary state factors (rectangles) and the possible actions (ellipsoids). Actions are red if they do not affect the environment, and blue if they result in the robot advancing to the next phase. The yellow action influences the yellow state, as shown with the yellow arrow.

established a fixed set of capabilities for the human by creating a secondary task that limits the human ability to act and observe the environment in Section III-A2. Fig. 1 shows an overview of the whole setup.

1) State-Action Space (Robot Capabilities): To meet **R2**, the state-action space used by the RL algorithm should be designed such that it enables a sufficient learning pace. RL algorithms usually require a great amount of training iterations, which is not possible if they need to learn alongside their human team partner. We have reduced the number of training iterations necessary by modeling the task through a state-action space that is as small as possible. This makes it possible to quickly explore all possible state-action pairs, also enabling relearning and therefore **R4**.

The actions modeled are a set of seven predetermined movements, and the states are determined by a set of four binary state factors, visualized in Fig. 2. The state factors mostly contain information about the human team member. Therefore, they provide the robot with some transparency of the human (**R5**). Moreover, the handover task is broken up into three phases. In each phase, only one or two of the state factors are taken into consideration, thereby effectively breaking the task up into three separate learning problems to further reduce the complexity. In each phase, the robot has different actions that it can choose from, as shown in Fig. 2.

Phase 1 describes the start of the handover, during which the robot needs to learn when to start handing over the object. The robot can observe whether the hand of the human team member is in the robot's workspace. The robot has two available actions: waiting until the state changes, or moving the object towards the human with the action *Go to human*.

When the robot takes the action (*Go to human*), it moves to Phase 2, during which the robot is moving towards the human. While moving, it can decide on the orientation it will use to hand over the object. The state factor that can be observed is the orientation of the human hand. The robot can choose between two predetermined orientations: the palm of the robot hand facing up (*Serve*), so that the human can take it out, or the palm facing down (*Drop*), to drop the object into the human hand.

After either action in Phase 2, the robot will move to Phase 3, in which the focus is on the handover. The robot needs to learn when and how far to open its hand, while the human needs to grasp or catch the object to prevent it from falling. The human can influence the robot's behavior by pulling on the object and displacing the end-effector. An additional state factor describes whether the robot hand is still fully closed or partially opened. This combination of capabilities presents opportunities for multiple strategies and soft dependencies to emerge. For example, the robot can learn to wait until the end-effector is displaced before opening its hand to ensure that the human is already holding the object when the robot lets go, or to open its hand enough to allow the human to take the object out without dropping it.

2) Secondary Task (Human Capabilities): A secondary task was introduced for the human in the form of a game-like task, that could only be completed if the human successfully received the handover object. The introduction of a secondary task gives the human incentive to complete the task by rewarding the human for the collaborative performance, to ensure a shared goal (**R1**). It also creates a motive for the human to get the object handed over from the robot, thereby creating a hard dependency (**R3**). Moreover, the secondary task could compensate for the superior ability of the human to observe task, to prevent a transparency

inequality (**R5**) and give control over the capabilities of the human.

The secondary task we designed required the human to track an asteroid on a screen. For that, they needed to continuously have one hand on a mouse and constantly have their eyes on the screen. They were instructed to deflect the asteroid to complete the game, and that for this to be possible, they needed a physical object that served as a projectile. They could not get up and get the object themselves, as they needed to keep tracking the asteroid on the screen. The task consisted of two stages. During the first stage, the human needed to keep their second hand on a button, until a loading bar was filled. As soon as the bar was full, they could let go of the button, while a timer started to count down, starting the second stage. In this stage, the human had twenty seconds to receive the object from the robot with their free hand, while still tracking the target on the screen with the other. When the team succeeded, the human was rewarded with the same score as the robot. If the task failed, both were rewarded negatively. Auditory feedback was provided in addition to reward screens, to engage the human.

B. Robot Reinforcement Learning Algorithm

After comparison of multiple RL algorithms [24], [25], [26] and their suitability for embodied human-robot co-learning applications, we chose to extend and adapt a Q-learning algorithm [27]. Q-learning is a simple and robust RL technique, that is often used in the domain of social robotics [28], [29], [30] and specifically co-learning [3]. We adapted the Q-learning algorithm using decomposition techniques based on MAXQ value decomposition [31] and extended it with eligibility traces [32] to specifically meet our design requirements.

1) Decomposition: Hierarchical RL with MAXQ Value Function Decomposition [33] decomposes the learning problem into multiple smaller problems with a hierarchical structure, resulting in faster learning [34]. Splitting the problem into smaller problems can also increase adaptability [31], as the policy of one phase of the learning problem can change without affecting the policies of other phases.

The idea of decomposing the problem is based on the concept that not every state variable is important in every phase of the task. The three phases in our task (see Fig. 2) are however sequential instead of hierarchical, meaning that they can not be decomposed using Dietterich's hierarchical value decomposition [33]. Therefore, we instead decomposed the learning problem into three sequential Q-learning problems, each with their own Q-table, creating the same effect of decreasing the amount of Q-values without affecting the amount of actions and state variables. The Q-values are thus a function of state (s), action (a) as well as phase (ϕ):

$$Q_*(\phi, s, a) = E\left[R(\phi', s') + \gamma \max_{a'} Q_*(\phi', s', a')\right]. \quad (1)$$

By decomposing the task, we provided the robot with information about the importance of state variables in different phases of the task. This significantly reduces the number of states and thereby decreases the scale of the learning problem, increasing the overall learning pace and adaptability of the RL agent, ensuring **R2** and **R4**.

2) Reward Function: Design requirement **R1** states that both agents get rewarded based on performance, and that both agents get rewarded similarly. Both agents therefore received either positive or negative feedback at the end of each episode. This reward was based on whether the handover was completed successfully without dropping the object, as well as the time left to do so. The robot received a positive reward of $(+10)$ if the task was completed successfully, and a negative of (-10) if the task failed. Additionally, if the task succeeded, the amount of seconds left to complete the task was added to the positive reward. As the team was given 20 seconds at the start of the task, the positive reward was always between $+10$ and $+30$. The reward function was extended with a small punishment (-1) for each action necessary to prevent a policy where the robot gets stuck in a loop. To accommodate **R1**, the human would see this same reward as a score given for the completion of the task.

3) Eligibility Traces: Rewarding the Q-learning algorithm at the end of each episode creates two problems. First of all, most actions get a delayed reward [35], making the learning pace slow. Second, due to the decomposition that we implemented, the Markov property is not satisfied across the whole task (as the task is decomposed into separate learning problems). Therefore, regular backpropagation does not work as effectively as it would otherwise. We solved these problems with eligibility traces [32].

An eligibility trace is a trace of all previously visited Q-values. These traces are stored in a table for each state-action pair in each phase $S(\phi, s, a)$. Using eligibility traces, the algorithm tracks all state-action pairs reached during the episode. When a reward is received at the end of an episode, it updates all corresponding Q-values based on this reward. This not only speeds up the learning process, but it also ensures that mistakes made in early phases of the task get rewarded negatively in case of an unsuccessful episode [36].

With eligibility-traces, all Q-values are updated after every action. To do so, we first calculate what would have been the updated Q-value for the last phase-state-action combination $\hat{Q}(\phi, s, a)$ shown in (2a) using the decomposed Bellmann (1). Then we use $\hat{Q}$ to calculated the update-value Δ_Q (2b):

$$\hat{Q}(\phi, s, a) = R(\phi', s') + \gamma \max_{a'} Q(\phi, s', a'), \quad (2a)$$

$$\Delta_Q = Q(\phi, s, a) - \hat{Q}(\phi, s, a). \quad (2b)$$

This update-value (Δ_Q) is then used to update all Q-values based on their eligibility. As shown in (3):

$$Q(\phi, s, a) = Q(\phi, s, a) + \alpha \Delta_Q S(\phi, s, a) \quad \forall \; S(\phi, s, a). \quad (3)$$

The learning rate α is a value between 1 and 0. It is used to determine to what extent new experiences override what has been learned already.

4) Epsilon Decay: The algorithm uses epsilon decay to balance exploration and exploitation. Usually, methods for balancing exploration and exploitation are designed to converge to a greedy policy. This is not beneficial for adaptability in the later stages of the learning process. In our algorithm, ϵ never

completely decays to zero. This enhances **R4** as the system must never stop exploring to stay adaptable during all stages of the learning process.

The reward (R) and the epsilon decay rate (γ_ϵ) are used to update ϵ as follows:

$$\epsilon = \begin{cases} \max(\gamma_\epsilon \epsilon, 0.2) & \text{if } R < 0 \vee \epsilon > 0.5 \\ \min(\frac{1}{\gamma_\epsilon}\epsilon, 0.5) & \text{if } R > 0 \end{cases}. \quad (4)$$

Epsilon starts at a value of 1 to guarantee exploration when no policy is learned yet. Epsilon then decays until it reaches a 50% change of exploration ($\gamma_\epsilon = 0.9$). During the rest of the episodes, ϵ decays further when the team has a high success rate, so the robot has a higher chance to exploit its current policy. When the team experiences more failure, the chance of exploring grows.

C. Evaluation

To test whether the designed task and robot algorithm would allow for co-learning, we evaluated the setup with human participants using a qualitative, case-by-case analysis. Co-learning is an open-ended process in which good task performance can manifest in many different ways, due to the multiple possible strategies that can be taken by the team. A qualitative approach can provide 1) an understanding of how individual co-learning behaviors evolve over time, and 2) insight into how our proposed new co-learning task provides an environment to hypothesize and quantify co-learning processes in the future. We aim to contribute to existing work on Co-Learning that also takes a qualitative approach [6], [12]. Six human participants (students from Master programs at Delft University of Technology) each performed the task for four sessions of ten minutes (40 minutes of co-learning per dyad in total). This resulted in approximately 20 to 30 handover attempts per session. Participants received written information about the procedure beforehand, and were further instructed verbally. We used six measurements for our evaluation: performance, subjective Collaboration Fluency, behavioral strategies, relative liability, action preference from Q-values, and answers to interview questions. The procedure was approved by the Human Research Ethics Committee at Delft University of Technology.

1) Performance: To track performance over time, we stored whether each attempt of the task was successful or not, and calculated the percentage of successful attempts per every ten-minute session.

2) Subjective Collaboration Fluency: To measure how the human participants experienced the collaboration, they were asked to complete a survey on human-robot Collaboration Fluency [37] after each ten-minute session.

3) Behavioral Strategies: As described earlier in the letter, there are several possible strategies that all lead to a successful handover. Considering the task design, we identified three possible strategies:

S1: The robot lets the object go, trusting the human will catch it.
S2: The human pulls on the object, letting the robot know it can let go.
S3: The robot opens its hand partially, letting the human take the object.

We recorded videos of the participants, such that we could qualitatively assess which strategies were followed in successful attempts.

4) Relative Liability: Relative liability describes the proportion in which team members caused episodes to fail in each 10-minute session. It portrays the relative learning pace of both agents, since when the learning pace is similar, it should stay the same for both agents over time. If one team member learns faster than the other, there is a shift in relative liability because the proportion of mistakes made by the superior agent goes down. We determined relative liability by checking which agent made the mistake that caused the episode to fail in the case of a failed episode.

We considered the robot to be responsible for failure when the object was not passed within the allocated time when the human did try to signal the robot, or when the robot dropped the object without the human touching it. For any other reason of failure, we considered the human liable.

5) Action Preference From Q-Values: Action preference describes the specific policy of the robot in different phases of the task. We have evaluated the Q-tables after each episode, to track which action the robot preferred in each state and whether and how this changed as the experiment progressed. This gave us insight into the behavior learned by the robot, as well as how adaptable this behavior is (how much it changes over time).

6) Interview: We conducted a short interview with each of the participants after the last learning, in which we asked the following three questions:

Q1: Please indicate what your objective was during the learning process.
Q2: Describe the different strategies that you used, and how did this change over time.
Q2: Did you rely on a specific strategy of the robot?

The first question was asked to investigate whether the goal of the human corresponded to the goal of the robot, to test whether the team had a shared goal **R1**. The second question was used to find if the human explored different strategies during the learning process, and more specifically whether it converged towards preferring one strategy over other strategies. With the last question, we intended to find whether the human experienced soft dependencies.

IV. Results

We present the results of the evaluation by analyzing whether we succeeded in facilitating the design requirements for each human-robot dyad (summarized in Table I).

A. Shared Goal (R1)

Participants B, C, E, and F indicated in the interview that their goal was to complete each episode without dropping the object. Participants B and F even indicated that they had a secondary goal of improving the time in which they succeeded, to optimize

TABLE I
OVERVIEW OF EACH REQUIREMENT AND WHETHER IT WAS MET DURING THE EXPERIMENT IN EACH TEAM

Teams	A	B	C	D	E	F
R1 - Shared Goal	X	✓	✓	-	✓	✓
R2 - Synchrony	✓	✓	-	X	✓	✓
R3 - Interdependence	✓	✓	✓	✓	-	✓
R4 - Adaptability	✓	✓	-	✓	-	✓
R5 - Transparency	✓	✓	X	-	✓	✓
Co-learning	✓	✓	-	-	-	✓

Additionally, the bottom row shows whether the results indicate that co-learning took place during the experiment. The content of the bottom row is discussed in Section V.

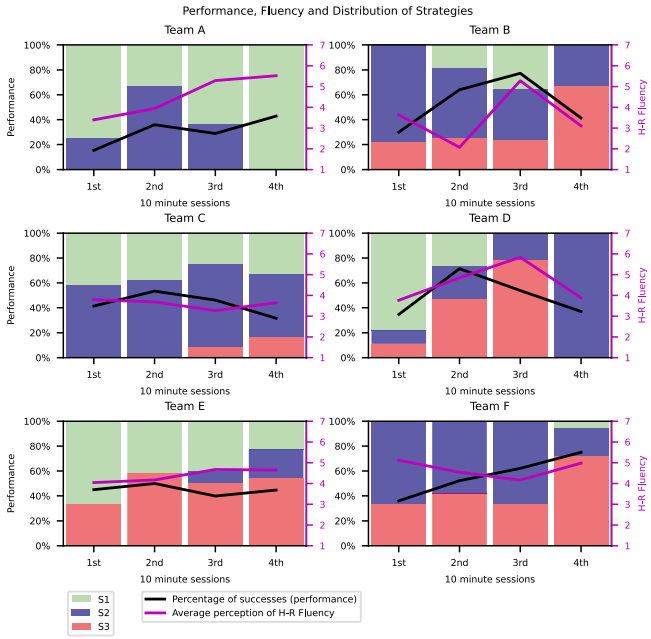

Fig. 3. The distribution of the three different strategies in phase 3 that could lead to a successful handover, combined with the performance and Collaboration Fluency score. The figure shows how the preference for different strategies changes over time. The three strategies are as follows: S1: The robot lets go of the object, trusting the human will catch it. S2: The human pulls on the object, letting the robot know it can let go. S3: The robot opens its hand partially, letting the human take the object.

their score. Therefore, it is shown in Table I that requirement **R1** is met in these teams.

Participants A and D indicated that their main goal was not to succeed at the task, but to train the robot to follow their preferred strategy. Participant A said that this was their main objective during the whole experiment. For instance, they never let the task succeed if the robot did not let go of the object. This is why strategy **S3** is never seen in team A in Fig. 3. Additionally, it can be seen that the human's perception of Fluency is relatively high, while the team's performance is low. This can be explained by the fact that the human met their objective of influencing the robot's behavior, at the expense of the robot's goal of succeeding at the task. This resulted in both agents perceiving different rewards.

Participant D indicated that they changed their objective between session three and four. First, it matched the robot's objective, while during the last session their goal was only to

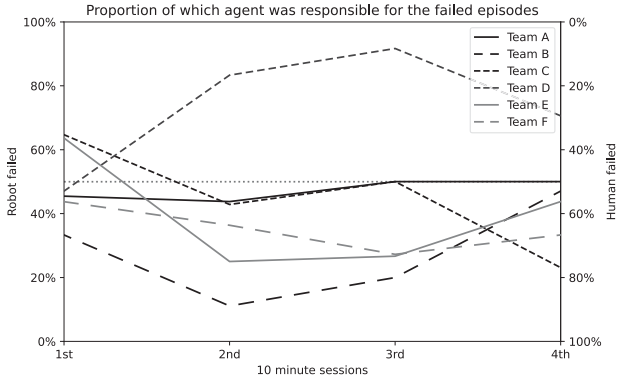

Fig. 4. The percentage of how many times each agent was responsible for failing an episode during each 10-minute session is shown for each team (relative liability). The rate of failure of the robot can be read on the left axis, while the human failure rate is displayed on the right axis.

train the robot to their preferred strategy. Participant D explained that their goal changed due to the realization that the robot used trial-and-error learning. They realized that they could influence the robot's behavior by rewarding desired behavior and punishing undesired behavior. Participant D therefore suddenly changed their behavior, resulting in the performance drop in the 4th session (Fig. 3) and the human deliberately failing the task to train the robot (Fig. 4). This behavior resulted in the team having a shared goal for only a part of the task, leaving **R1** inconclusive in team D.

B. *Synchrony (R2)*

Fig. 4 shows that relative liability is relatively constant for teams A and F, which means that the learning of the human and the robot was synchronous. For teams B and E, there is an initial large shift towards the human being responsible for a large part of the failures, but this recovers to the middle over subsequent sessions. While the robot learned faster in the beginning, the human was able to catch up, leading to synchronous learning overall.

In team D we can observe a shift in relative liability in the first 3 sessions, as the robot could not keep up with the learning of the human. Therefore, this synchrony requirement was not met. Moreover, the human still learned faster than the robot during the fourth session, even though Fig. 4 seems to imply that the robot made a recovery. The reason this proportion drops back towards 50%, however, is that the human started to deliberately fail the task to actively train the robot to prefer strategy **S2**, as mentioned by the participant in the interview. This can also be seen by the sudden decrease in performance rate during this session in Fig. 3.

In team C, a shift in relative liability can be seen in Fig. 4. In this case, the robot improved its policy faster than the human. Combined with the fact that the team barely improved their performance during the four sessions (see Fig. 3), we can deduce that the human did not improve its policy at all. Therefore, no conclusion could be drawn about this requirement for this team.

C. Interdependence (R3)

Different individuals prefer different strategies. Fig. 3 shows that the method enables different teams to learn different strategies. Team A, for instance, converges completely to strategy **S1**, while teams B and D learned that this strategy did not work for them.

Post-experiment interviews revealed some important underlying insights about the development of the strategies during co-learning. Participants A and C stated that they did not want to take the object from the robot without its permission, in an attempt to maintain the trust of the robot. This complies with the action preferences from the Q-values, visualized in Fig. 3 that shows that strategy **S3** was not preferred in these teams. By actively not choosing this strategy, the human depends on the robot to open its hand completely for the task to be completed. This shows the establishment of a soft dependency between the human and the robot, which is beneficial for the team's relationship.

All three observed strategies contain their own similar soft dependencies. This means that soft dependencies arise during the learning process, when a team converges to preferring one specific strategy. It can be seen in Fig. 3 that in all teams multiple strategies were explored. In teams A, B, D, and F, there was convergence to one specific strategy during the experiment. Thus, soft dependencies emerged in these teams. Team E is the only team that kept executing all strategies until the end of the experiment. This means that both team members never fully committed to being dependent on the other, making it the only team for which it is inconclusive whether design requirement **R3** is met.

D. Adaptability (R4)

Fig. 3 clearly shows that teams A, B, D, and F made a relatively drastic change in preferred strategy towards the end of the experiment. This shows that the RL algorithm was able to adapt its policy to accommodate strategy changes.

In teams C and E, no large change in the policy of the robot occurred during the experiment. However, this does not necessarily mean that the robot had no adaptability, as the result could have also been caused by behavior of the human. Therefore, these cells are inconclusive in Table I.

E. Transparency (R5)

The method allowed both agents to observe each other by design, as explained earlier in the letter. We attempted to avoid an imbalance in learning pace by ensuring that both human and robot had access to a similar level of limited information about the other.

Fig. 4 shows that there was indeed no imbalance in learning pace in teams A, B, E, and F, as explained in Section IV-B. The unequal learning pace in team C, however, was caused by the fact that the human was not able to understand the policy of the robot. This was a result of the human being too occupied by the secondary task, resulting in unbalanced transparency. There is no indication that the unequal learning pace in team D was caused by the same issue.

While the secondary task prevented the human from constantly looking at the robot, as explained in Section III-A2, Fig. 3 shows that in teams B, D, and F, the human preferred to rely on tactile sensing to know where to grasp the object, as they do not follow strategy **S1**. They were therefore able to compensate for their lack of visual observation. Further investigation of the video recordings of the experiment showed that participants A and E also relied on tactile sensing to locate the object, they just did it in a subtle manner, so as to not displace the robot.

In short, in teams A, B, E, and F we can state that both agents had transparency and that no unwanted imbalance appeared. This means the requirement is met. In team C, the secondary task over-hindered the human's ability to visually observe the robot, causing this requirement not to be met, while in team D the results are inconclusive.

V. Discussion

A. Facilitating Co-Learning

In summary, in three out of six teams we managed to create the circumstances for Co-Learning. Table I shows that all the requirements are met in team B and team F, meaning that these teams demonstrated successful Co-Learning. Nevertheless, partially fulfilling the requirements can still mean that there was some degree of Co-Learning, as shown by the development of interesting and useful human-robot collaboration patterns. For example, even though the human and the robot did not have the same goal (**R1**) in team A, they still managed to improve their collaboration by co-learning a joint strategy. The reason **R1** was not met in team A, is that the participant chose to train the robot, while the goal of the robot was to succeed at the task. In practice, however, there are still multiple congruent strategies that reach both goals. Moreover, Fig. 3 shows an increase in performance over time for team A, as well as a growth in the participant's perception of fluency in the team. The better team performance may be the result of effective collaboration patterns, and the improved fluency suggests that soft dependencies emerged in team A.

In team C, we observed that the participant struggled to understand how to do the task and seemed unable to learn this within the given time. Still, the team developed some collaboration patterns and soft dependencies. However, by not meeting **R2**, **R4** and **R5** we cannot claim that this team was able to achieve a full Co-Learning process.

In team D, the human learned faster than the robot, which led to an imbalance in contribution over time (Fig. 4). This imbalance may have caused the human to change their motivation over time. Even though there were indications for co-learning during the first sessions of the experiment, it was not sustained during the last session. Therefore, in team D, multiple design requirements were left inconclusive or were not met.

In Team E, we did not see soft dependencies arise during the experiment. Additionally, Fig. 3 does not show an increase in performance or fluency. However, changes in preferred collaboration patterns over time can still be observed in Fig. 3. They

are not substantial enough to prove that **R3** or **R4** were met, but it does suggest that a longer collaboration could have resulted in the requirements being met. Furthermore, Fig. 4 shows a balanced learning pace between the two agents. Overall, it seems that the team was still exploring after the four sessions, and full Co-Learning might have emerged in a longer collaboration.

B. Limitations and Future Work

While the results showed that the developed task-, algorithm- and interaction design enabled co-learning, the short-term performance was not increased for most teams (Fig. 3). This can be explained by the fact that the designed method and evaluation focused on the Co-Learning process. The essence of designing for Co-Learning is to design the conditions in which collaborators can learn the behavior needed to develop smooth and effective collaboration, which can facilitate long-term performance improvement. This means that Co-Learning can be present without an immediate performance increase. We expect that when a similar experiment is done for a longer duration of time, with more participants, an increase in performance should be measurable in teams where Co-Learning is identified. The work presented in this letter can serve as a basis for future larger, quantitative studies into long-term effects of Co-Learning in embodied human-robot teams.

This study focused on facilitating human-robot Co-Learning through five requirements. As we were able to identify co-learning in at least one of the teams (team A) despite one of the requirements not being met, the individual effect and weight of each design requirement requires further research.

REFERENCES

[1] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Auton. Robots*, vol. 42, no. 5, pp. 957–975, 2018.

[2] K. van den Bosch, T. Schoonderwoerd, R. Blankendaal, and M. Neerincx, *Six Challenges for Human-AI Co-learning*, (Lecture Notes in Computer Science Series), vol. 11597. Cham, Switzerland: Springer, 2019.

[3] E. M. van Zoelen, K. van den Bosch, and M. Neerincx, "Becoming team members: Identifying interaction patterns of mutual adaptation for human-robot co-learning," *Front. Robot. AI*, vol. 8, 2021, Art. no. 692811.

[4] E. M. van Zoelen, K. van den Bosch, M. Rauterberg, E. Barakova, and M. Neerincx, "Identifying interaction patterns of tangible co-adaptations in human-robot team behaviors," *Front. Psychol.*, vol. 12, 2021, Art. no. 645545.

[5] Y. Li, Z. Zhang, and F. Zhang, "Hybrid co-learning for proximate human-robot teaming," in *Proc. IEEE 20th Int. Conf. Ubiquitous Robots*, 2023, pp. 239–244.

[6] A. Shafti, J. Tjomsland, W. Dudley, and A. A. Faisal, "Real-world human-robot collaborative reinforcement learning," in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 11161–11166.

[7] M. Johnson, J. M. Bradshaw, P. J. Feltovich, C. M. Jonker, M. B. van Riemsdijk, and M. Sierhuis, "Coactive design: Designing support for interdependence in joint activity," *J. Human-Robot Interact.*, vol. 3, no. 1, pp. 43–69, Feb. 2014.

[8] S. Ikemoto, H. B. Amor, T. Minato, B. Jung, and H. Ishiguro, "Physical human-robot interaction: Mutual learning and adaptation," *IEEE Robot. Automat. Mag.*, vol. 19, no. 4, pp. 24–35, Dec. 2012.

[9] L. Peternel, E. Oztop, and J. Babič, "A shared control method for online human-in-the-loop robot learning based on locally weighted regression," in *Proc. 2016 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 3900–3906.

[10] S. Nikolaidis, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in collaborative tasks: Models and experiments," *Int. J. Robot. Res.*, vol. 36, no. 5/7, pp. 618–634, 2017.

[11] N. Amirshirzad, A. Kumru, and E. Oztop, "Human adaptation to human–robot shared control," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 2, pp. 126–136, Apr. 2019.

[12] R. Kumar, Y. Huan, and N. Yao, "Unveiling the dynamics of human decision-making: From strategies to false beliefs in collaborative human-robot co-learning tasks," in *Proc. Companion 2024 ACM/IEEE Int. Conf. Hum.-Robot Interact.*, 2024, pp. 632–636.

[13] H. Loopik, "Comparing reinforcement learning techniques for embodied co-learning in a human-robot team," Technische Universiteit Delft, Tech. Rep., 2022, MSc Literature Study. [Online]. Available: https://github.com/HugoLoopik/Co-learning

[14] T. Y. Katz-Navon and M. Erez, "When collective- and self-efficacy affect team performance: The role of task interdependence," *Small Group Res.*, vol. 36, no. 4, pp. 437–465, 2005.

[15] R. Rijgersberg-Peters, W. van Vught, J. Broekens, and M. A. Neerincx, "Goal ontology for personalised learning and its implementation in child's health self-management support," *IEEE Trans. Learn. Technol.*, vol. 17, pp. 903–918, 2024.

[16] A. Mörtl, T. Lorenz, and S. Hirche, "Rhythm patterns interaction-synchronization behavior for human-robot joint action," *PLoS One*, vol. 9, no. 4, 2014, Art. no. e95195.

[17] S. C. F. Hendrikse, "Interpersonal synchrony: Analyzing and modeling social interaction dynamics," PhD Thesis, Vrije Universiteit Amsterdam, Tech. Rep., 2024.

[18] H. Doorewaard, G. van Hootegem, and R. Huys, "Team responsibility structure and team performance," *Personnel Rev.*, vol. 31, no. 3, pp. 356–370, Jun. 2002.

[19] C. S. Burke, K. C. Stagl, E. Salas, L. Pierce, and D. Kendall, "Understanding team adaptation: A conceptual analysis and model," *J. Appl. Psychol.*, vol. 91, no. 6, 2006, Art. no. 1189.

[20] M. Vössing, N. Kühl, M. Lind, and G. Satzger, "Designing transparency for effective human-ai collaboration," *Inf. Syst. Front.*, vol. 24, no. 3, pp. 877–895, 2022.

[21] K. Haresamudram, S. Larsson, and F. Heintz, "Three levels of ai transparency," *Computer*, vol. 56, no. 2, pp. 93–100, 2023.

[22] R. S. Verhagen, M. A. Neerincx, and M. L. Tielman, "The influence of interdependence and a transparent or explainable communication style on human-robot teamwork," *Front. Robot. AI*, vol. 9, 2022, Art. no. 993997.

[23] V. Ortenzi, A. Cosgun, T. Pardi, W. P. Chan, E. Croft, and D. Kulic, "Object handovers: A review for robotics," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1855–1873, Dec. 2021.

[24] N. Akalin and A. Loutfi, "Reinforcement learning approaches in social robotics," *Sensors*, vol. 21, no. 4, 2021, Art. no. 1292.

[25] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1861–1870.

[26] S. Ramstedt and C. J. Pal, "Real-time reinforcement learning," 2019, *arXiv:1911.04448*.

[27] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279–292, May 1992.

[28] C. Moro, G. Nejat, and A. Mihailidis, "Learning and personalizing socially assistive robot behaviors to aid with activities of daily living," *J. Hum.-Robot Interact.*, vol. 7, no. 2, pp. 1–25, Oct. 2018.

[29] I. Papaioannou, C. Dondrup, J. Novikova, and O. Lemon, "Hybrid chat and task dialogue for more engaging HRI using reinforcement learning," in *Proc. 26th IEEE Int. Symp. Robot Hum. Interactive Commun.*, 2017, pp. 593–598.

[30] J. Hemminahaus and S. Kopp, "Towards adaptive social behavior generation for assistive robots using reinforcement learning," in *Proc. 2017 12th ACM/IEEE Int. Conf. Hum.-Robot Interact.*, 2017, pp. 332–340.

[31] T. G. Dietterich, "Hierarchical reinforcement learning with the MAXQ value function decomposition," *J. Artif. Intell. Res.*, vol. 13, pp. 227–303, 2000.

[32] R. S. Sutton, "Temporal credit assignment in reinforcement learning," Ph.D. dissertation, Univ. Massachusetts, Amherst, 1984.

[33] T. G. Dietterich et al., "The MAXQ method for hierarchical reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 1998, pp. 118–126.

[34] J. Chan and G. Nejat, "Social intelligence for a robot engaging people in cognitive training activities," *Int. J. Adv. Robot. Syst.*, vol. 9, no. 4, 2012, Art. no. 113.

[35] R. S. Sutton, "Introduction: The challenge of reinforcement learning," in *Reinforcement Learning*, Berlin, Germany: Springer, 1992, pp. 1–3.

[36] S. P. Singh and R. S. Sutton, "Reinforcement learning with replacing eligibility traces," *Mach. Learn.*, vol. 22, no. 1–3, pp. 123–158, 1996.

[37] G. Hoffman, "Evaluating fluency in human–robot collaboration," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 3, pp. 209–218, Jun. 2019.