# Play-to-Coach: Coaching Humans to Play Air Hockey with Robots

***Abstract*—Robots are increasingly assuming diverse roles in physical interaction tasks with humans, evolving from mere learners and collaborators to potential teachers in tasks requiring physical skills. This shift is crucial in scenarios where human teachers are scarce, establishing robots as an effective means to augment teaching. Against this backdrop, this paper introduces Play-to-Coach, a robot system built to coach humans in physical tasks, focusing on air hockey puck repelling. Play-to-Coach combines skill decomposition and adaptive teaching strategies. Skill decomposition breaks down the complex task into more manageable sub-skills for effective learning. Concurrently, adaptive teaching, steered by a Multi-Armed Bandit algorithm, flexibly adjusts the sequence of sub-skills in response to the learner's progression. Human subject experiments validate Play-to-Coach's effectiveness and discern which system features most effectively facilitate skill learning outcomes and experience.**

## I. Introduction

Robots have significantly expanded their roles in various physical interaction tasks with humans, evolving from learners of human behaviors, collaborators, and assistants to potential teachers in physical tasks [1, 2, 3, 4, 5, 6, 7]. This evolution is particularly crucial in contexts with a scarcity of human teachers or experts, positioning robots as a viable solution for scaling up teaching efforts, especially in physical tasks where one-on-one interaction is key for effective learning. However, the task of teaching humans physical skills via robots introduces several challenges, mainly due to the complexity and continuous nature of these skills' input and output spaces. There is also the need to adapt effectively to diverse human learning styles and physical capabilities while maintaining learner engagement and motivation. This leads to our research question: *"How can robots autonomously coach humans in two-player physical tasks, ensuring both good learning outcomes and a positive learning experience?"*

This paper focuses on a two-player physical task, specifically air hockey. Such a class of tasks presents unique challenges compared to single-player tasks. Existing approaches [8] to teaching these skills often rely on segmenting expert demonstrations for learners to imitate, a method that becomes increasingly complicated and resource-intensive in multiplayer settings. Firstly, the collection of demonstrations would necessitate the coordination of multiple participants. Secondly, direct imitation may not be an optimal approach for all learners, due to variations in individual physical capabilities and the complexities of multiplayer dynamics.

To overcome these limitations, we present Play-to-Coach, a robot system designed to coach humans in the skill of repelling a puck in air hockey. The robot coaches the human learner by playing with him/her, thus Play-to-Coach, shifting from direct imitation to collaborative learning. The premise of the system is straightforward: the robot serves the puck from a fixed position to the human player, who then learns to repel the puck into the goal from different angles that mimic various real-world sports training scenarios, such as badminton and table tennis. The primary goal of Play-to-Coach is to provide the human learner with an effective puck-repelling skill, as illustrated in Fig. 1. At the heart of Play-to-Coach are two methods informed by previous research: *skill decomposition* [8, 7, 9] with learned expert policies and *adaptive teaching* [10, 11, 12].

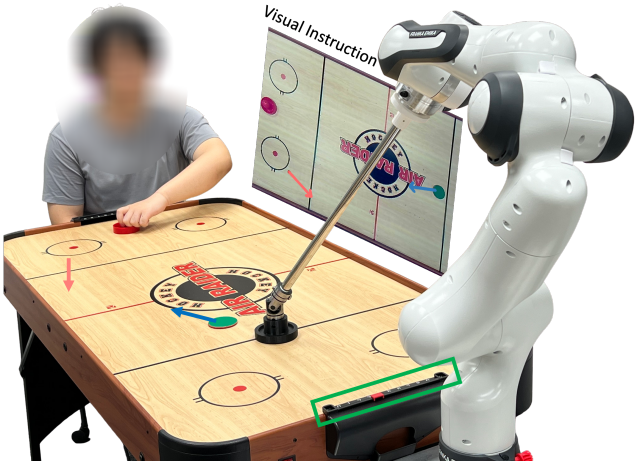



Fig. 1: A human learner learns to repel the puck into the goal region with a robot coach. The goal region is located in the middle of two ends of the table, bounded by the green box. In our Play-to-Coach system, the human learner is given visual instruction on how to repel the puck.

Skill decomposition aims to decompose complex physical skills into more manageable sub-skills for human learning [8]. The skill decomposition process in Play-to-Coach begins with training a puck-repelling policy using constrained reinforcement learning [13]. This specialized form of learning is crucial to ensure that the actions generated by the policy are realistic and can be replicated by human learners. From this trained policy, we generate diverse puck-repelling trajectories. These are further decomposed into smaller, teachable sub-skills through skill discovery [14]. Each sub-skill is defined by a pair of puck states, detailing both how the puck moves toward the player and the corresponding way for repelling it. During the teaching round, the human learner is given visual

instruction on how to repel it, followed by the robot coach delivering the puck in the corresponding direction.

Complementing the skill decomposition, Play-to-Coach integrates a Multi-Armed Bandit (MAB) algorithm [15] to optimize the teaching sequence. This algorithm selects a suitable sub-skill to teach at each learning round of interaction, guided by the learner's ongoing progress. By employing this algorithm, Play-to-Coach dynamically adapts the teaching sequence, enhancing learning efficiency and experience.

The effectiveness of Play-to-Coach was evaluated through a human subjects experiment. Participants are divided into four groups to interact with different variants of the Play-to-Coach. The experiment not only demonstrates the overall effectiveness of Play-to-Coach but also identifies which system features most effectively support skill learning and enhance user experience. Key findings include:

- Skill decomposition significantly improved learning outcomes.
- Adaptive teaching, while not substantially enhancing learning outcomes, was effective in reducing perceived workload and building trust.
- Learner behavior was strongly influenced by their trust in the learning outcomes and instructions provided by the robot coach.

These results emphasize the importance of human trust, the adaptability of sub-skills in physical robot coaching, and the need for improved communication from the robot coach.

## II. Related Works

### A. Robot Learning

The field of robot learning, which integrates machine learning techniques into robotic systems, has garnered significant attention due to its potential to enhance robot capabilities across various tasks. This burgeoning interest is evident in areas such as robotic manipulation [16, 17, 18, 19], navigation [20], and human-robot interactions [21]. A pivotal aspect of this field is the direction of knowledge transfer, primarily from humans to robots, symbolizing a traditional student-teacher mode where the robot assumes the role of the learner. Techniques such as imitation learning [22, 23] and reinforcement learning [24, 25] are central to this paradigm, with knowledge relies heavily on human inputs, either through reward design [26] or expert demonstrations [27, 28].

In this work, we diverge from this conventional framework by exploring the feasibility of reversing these roles – positioning the robot as the teacher and the human as the learner. This inversion of traditional knowledge flow still leverages established robot learning techniques. In Play-to-Coach, we demonstrate how robot learning can be utilized to empower the robot to assume the role of a teacher, thus opening new possibilities in the field of human-robot interaction.

### B. Intelligent Tutoring System

The use of machines for educational purposes has a long history, recently enhanced by advancements in deep learning. Intelligent tutoring systems, the early pioneers of automated teaching, have seen successful integration into many commercial applications [29]. Incorporating concepts like machine teaching [30, 31, 32, 33, 34] into these systems has shown promise in improving human learning efficiency. Although social robots have been explored for facilitating learning in various educational domains [35, 36, 37], teaching complex motor control skills through mere visual or verbal cues is still challenging, typically requiring extensive practice. Modern machine learning advancements enable more adaptive curricula and detailed feedback for teaching physical skills [8, 38]. Further, robot-assisted teaching has been introduced to provide explicit physical guidance in these tasks [39, 7].

Unlike existing intelligent tutoring systems, Play-to-Coach operates on a two-player physical tasks. It reduces the need for human expert involvement by autonomously identifying both the expert and the sub-skills needed to teach. In addition, the robot coaches the human learner by playing with him/her, thus Play-to-Coach shifts from direct imitation to collaborative learning to support more classes of tasks.

### C. Physical Human-Robot Interaction (pHRI)

In the realm of physical Human-Robot Interaction (pHRI), a key area of focus is how robots can assist humans in achieving their hidden objectives [40, 41]. The primary goal for the robot in such interactions is to deduce human intentions and adapt its assistance accordingly. Action selection and human intention inference are often considered separate processes [42, 43, 44]. To address this, a decision-theoretic framework like the Assistant Partially Observable Markov Decision Process (POMDP) has been developed, encapsulating the broader concept of assistance in pHRI [45]. In this framework, robots integrate reward learning and control modules for advanced reasoning over human feedback [46, 47]. Another crucial aspect of pHRI is modeling interactions as collaborative efforts between humans and robots, where both parties work towards a common goal [48, 4]. In such scenarios, the joint optimal policy, like rotating a table counter-clockwise, is initially unknown to both agents. Their interaction evolves through mutual adaptation [49, 50, 51]. Previous studies have shown that machine learning enables robots to steer human behavior towards a desired outcome [52, 53].

Our work focuses on a closely related but relatively under-explored setting, where the robot is not only assisting humans but also trying to teach humans certain skills physically. Such a task requires more than just influencing humans but may also require the robot to leverage expert knowledge to perform explicit teaching.

## III. System Overview

### A. Task Description

Our study is centered on an interactive air hockey game, where we have developed a robot system to act as a coach for human participants. In this setup, the human players are the "learners", and the robot assumes the role of the "coach". The robot coach is programmed to strike the puck with different velocities and trajectories, creating a dynamic and challenging

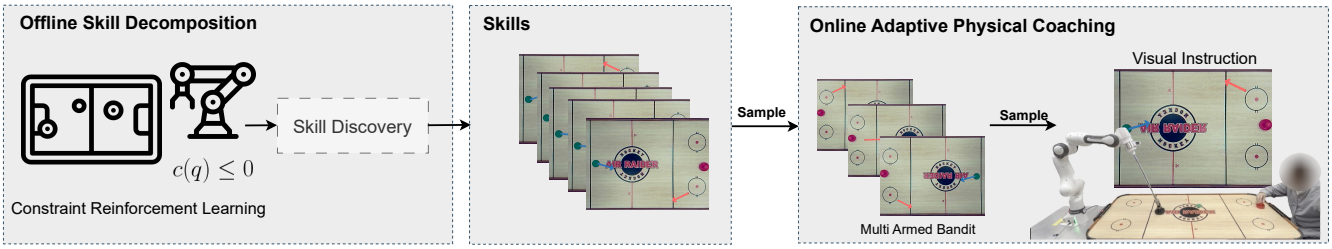



Fig. 2: The overall pipeline of Play-to-Coach. The process begins with training an expert policy in an air hockey simulator offline, followed by the decomposition of puck-repelling trajectories into sub-skills. Then a robot coach teaches these sub-skills to human learners through visual instruction and physical interaction. The red arrow indicates the "hitting state" while the blue arrow indicates the "initial state" in the visual instruction.

game environment. The learner's main goal is to repel the puck into the goal, adapting to its varying incoming trajectories.

During each game round, the robot coach initiates play by hitting the puck from a fixed position, thereby starting the interaction. The human learner then aims to repel the puck successfully into the goal in their defensive play. After each round, the puck is manually reset, with the interaction limited to one offensive move by the robot and a single defensive response by the human learner, who is allowed only one hit per round. Success in a round is defined by the puck being hit into the goal, while failure is marked by any other outcome. This game setting mirrors real-world coaching environments, challenging the learner with a specific task and requiring them to adapt to varying conditions set by the coach.

### B. Conceptual System Model

Conceptually, the problem can be divided into two parts, which establish a target task and a teaching task [54, 8, 7]. The target task is the original two-player task both the robot and human are involved. Specifically,

*Definition 1:* The *target task* is a two-player Markov game $\mathcal{M} = (S, A^1, A^2, T, R_1, R_2, \gamma)$ between two agents, 1 and 2, where

- $S$ is a set of target task states;
- $A^1$ is a set of actions for agent 1;
- $A^2$ is a set of actions for agent 2;
- $T(s'|s, a^1, a^2)$ is a conditional probability function on the next target task state $s' \in S$, given the current state $s \in S$ and both agents' actions $a^1 \in A^1$ and $a^2 \in A^2$;
- $R_1(s,\ a^1,\ a^2,\ s')$ is a target task reward function that maps the target task state and players' actions to a real number for agent 1;
- $R_2(s,\ a^1,\ a^2,\ s')$ is a target task reward function that maps the target task state and players' actions to a real number for agent 2;
- $\gamma$ is a discount factor.

At each step $t$, agent 1 and 2 observe the current task state $s_t$ and select their respective actions $a_t^1 \sim \pi^1$ and $a_t^2 \sim \pi^2$, where $\pi^i$ the agent's policy $i$ for $i = 1, 2$. They then receive the reward $r_t^1 = R_1(s_t, a_t^1, a_t^2, s_{t+1})$ and $r_t^2 = R_2(s_t, a_t^1, a_t^2, s_{t+1})$, respectively. The next state is updated as $s_{t+1} \sim T(s_{t+1} \mid s_t, a_t^1, a_t^2)$.

In the particular task of air hockey, we are interested in repelling the puck into the goal. We assume agent 2 is the agent who initializes the game by hitting the puck toward the opponent and agent 1 is the agent who aims to repel the puck into the goal. We define the state $s$ to contain two types of information, one is the environment state $s^e \in S^e$, which is mainly the pose and velocity of the puck, and the state of the agents who play the game, mainly the pose of the agents $q^1 \in Q^1, q^2 \in Q^2$, where $Q^1, Q^2$ are the configuration space of agent 1 and agent 2.

The next step is to define the teaching task. In this task, the learner is defined as a tuple of its policy, denoted by $\pi_l$, and how the policy is updated, which is represented by $U$. The policy takes the current environment state, denoted by $s^e$, as input and outputs an action. The update function, denoted by $U$, models how the learner changes its policy after each interaction. It is assumed that the history of observation of the state of the environment and the reward obtained by the learner at time step $t$ is $H_t = [(s_0^e, r_0^1), ..., (s_t^e, r_t^1)]$. Consequently, the student updates $\pi_l$ with any iterative functions conditioned on the history of interactions: $\pi_l^{t+1} = U(\pi_l^t, H_t)$.

*Definition 2:* Given a target task $\mathcal{M} = (S, A^1, A^2, T, R_1, R_2, \gamma)$, a learner $(\pi_l, U)$, and a policy to teach $\pi_l^*$ for the target task, the *teaching task* is a POMDP $\bar{\mathcal{M}} = (\bar{X}, \bar{A}, \bar{T}, \bar{O}, \bar{Z}, \bar{R}, \bar{\gamma})$ for the coach, where

- $\bar{X}$ is a set of teaching states: $\bar{x} = (s, \pi_l)$, for target task state $s \in S$ and learner policy $\pi_l$;
- $\bar{A}$ is a set of actions: $\bar{A} = A^1 \cup A^2$;
- $\bar{T}(\ \bar{x}' |\ \bar{x}, \bar{a})$ is a conditional probability function on the next state $\bar{x}' \in \bar{X}$, given the current state $\bar{x} \in \bar{X}$ and coach's action $\bar{a} \in \bar{A}$;
- $\bar{O}$ is a set of observations: $\bar{o} = (s, r)$, for target task state $s \in S$ and target task reward $r$;
- $\bar{Z}(\ \bar{o} \mid \bar{a}, \bar{x})$ is a conditional probability function on the observation $\bar{o} \in \bar{O}$ , given coach's action $\bar{a} \in \bar{A}$ and current state $\bar{x} \in \bar{X}$;
- $\bar{R}(\bar{x}, \bar{a}, \bar{x}')$ is a teaching reward function that maps current state $\bar{x} \in \bar{X}$, coach's action $\bar{a} \in \bar{A}$, and next state $\bar{x}' \in \bar{X}$ to a real number measuring the effectiveness of teaching;

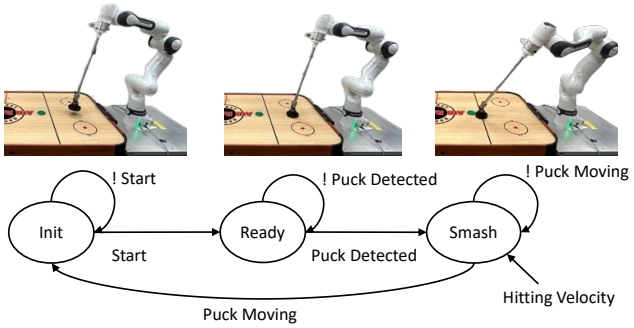



Fig. 3: Diagram of the high-level finite state machine for robot coach control during one round of the game.

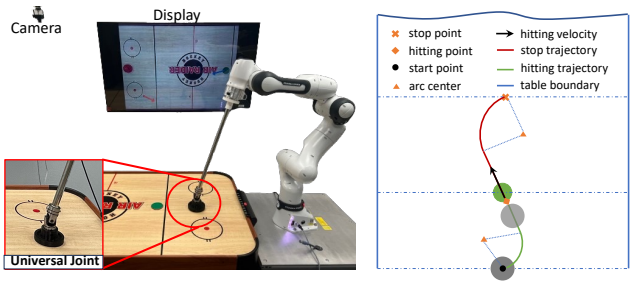



Fig. 4: Illustration of system setups and hitting path planning. A high-speed camera is positioned above to capture the trajectory of the puck. The robot coach offers visual instructions regarding the repelling directions through a display as a guidance to the learner.

- $\bar{\gamma}$ is a discount factor.

We call this POMDP the Teaching POMDP. The objective of this task is to enable humans to converge to the $\pi_l^*$ as fast as possible. Addressing the challenges in solving a Teaching POMDP for teaching physical skills involves two primary obstacles. Firstly, the extensive action space of physical tasks exacerbates the complexity of deriving an optimal solution. Secondly, the intricacies in modeling the transition function which is the human learning model in physical tasks, present significant difficulties. Data-driven methods of learning such a human model [39], while effective for specific tasks, often fall short in accurately modeling transitions across a diverse array of physical tasks due to data scarcity. This limitation hinders the reliability of long-term planning and searches for solutions prohibitively expensive due to the large action space.

To address these complexities, an effective strategy is the decomposition of the task into more manageable sub-tasks [8, 7, 55]. This not only facilitates the learning process for learners but also streamlines the decision-making process of teaching by focusing on discrete skills. Furthermore, adapting the POMDP framework to an MAB problem by leveraging empirical educational insights presents a viable solution [15]. For example, focusing on activities that provide more learning progress can act as a strong motivational signal for human learning [56] without knowing exactly what the human model is. Such heuristics can be easily encoded into the reward design in an MAB algorithm, thereby reducing dependence on comprehensive human cognitive models.

Building on these findings and prior research, the overview of Play-to-Coach, as illustrated in Fig. 2, is divided into three phases. The process begins with training a policy in an air hockey simulator using constrained reinforcement learning, which creates a virtual expert model. This model is then used to generate a variety of puck-repelling trajectories. Subsequently, these trajectories are broken down into smaller, more manageable sub-skills via skill decomposition methods. For this purpose, we utilize a Vector Quantized Variational Auto-Encoder (VQ-VAE) [57], which effectively captures discrete sub-skills similar to **(author?)** [14]. These sub-skills are identified by a pair of puck states: one detailing the initial state of the puck's movement towards the player, see Fig. 4 on how the puck is hit towards the player and the other describing the hitting state of the puck after it is reppelled. In the final phase, an MAB algorithm is employed to select which sub-skill to teach during each interaction with a human learner. During these teaching rounds, the learner receives visual instructions on the hitting state, after which the robot coach serves the puck toward them in the corresponding direction.

### C. System Setup

For the hardware setup of the system (see Fig. 4), we mounted one 7-DoF Franka Research 3 (FR3) robot at one end of an air hockey table and equipped each with a custom-designed end effector of length 520 mm. The end-effector was composed of an aluminum rod and a universal joint connected to a mallet, see Fig. 4 bottom left. Universal joint passively adapts the roll and pitch angles of the end effector to ensure that the mallet surface is parallel to the table. The cylindrical symmetry of the mallet warrants collisions which are invariant to yaw angles. Additionally, we positioned a high-speed camera with a resolution of $640 \times 480$ that runs at 240hz to capture the trajectory of the puck, see Fig. 4 top left. The display on the wall is used to provide visual instruction to human learners. The air hockey table we use is 1.6 meters in length ($x-$axis) and 0.72 meters in width ($y-$axis). The robot and human participant are positioned at opposite ends of the table, each approximately 1 meter from the table's center and aligned with the table's width. The puck is placed at a fixed position, 0.7 meters from the robot's side, and centered along the table's width. In our experiments, the robot initiates the game by hitting the puck with velocities ranging from [[0.5, 0.85], [-0.2, 0.2]] m/s, along $x-$axis and $y-$axis seen from the robot's coordinates.

On the software side, the hitting movement is the point-point movement planned in the workspace, the 2-D plane of the air hockey table. The hitting movement is implemented by the lower-level joint space position controller given reference position and velocity $\{q_r, \dot{q}_r\}$. We build the high-level control for skill teaching using a finite state machine similar to that in previous work [58]. In our particular implementation, we only use the *Init*, *Ready*, and *Smash* state. See the Fig. 3 for an

illustration. Each hitting round starts from the *Init* state, where the mallet positions at a safe height from the table. When the start command is active, the robot positions the mallet on the table and enters the *Ready* state. If the puck is detected and within the hitting zone, as shown in Fig. 4, the robot enters the *Smash* state. The hitting and stop points are calculated based on the desired hitting direction and velocity to teach specific skills. For each teaching round, a hitting movement consisting of a hitting trajectory and a stopping trajectory is planned in the Cartesian space based on two arcs given the start, hitting, and stop points. After hitting the puck, the robot will return to the *Init* state and wait for the renewed start command. FR3 has the extra redundancy for Cartesian trajectory tracking, the joint velocities can be calculated as $\dot{q}_r = \boldsymbol{J}^{\#}(q_r)\dot{x}_{ee} + \boldsymbol{N}\dot{q}_r$. $\dot{x}_{ee}$ is the desired Cartesian velocity, $\boldsymbol{J}^{\#}$ and $\boldsymbol{N}$ are the generalized inverse and the null space projection matrix. Therefore, the null-space joint velocities need to be optimized in real-time within the robot's joint maximum velocity constraints for hitting movement implementation. An Anchored Quadratic Programming (AQP), proposed by [58], is utilized to derive joint velocities and improve the hitting quality by adding an optimized hitting configuration as a reference along the hitting direction and position. Consequently, a real-time hitting trajectory of joint velocities without collisions that satisfies the constraints of the robot's physical and hitting movement is derived.

## IV. System Components

In this section, we elaborate on the process of acquiring sub-skills and detail the use of these skills within an MAB-based adaptive teaching algorithm.

### A. Expert Policy Acquisition

The initial phase of Play-to-Coach focuses on developing an expert policy, $\pi_\theta$, which serves as a proxy of $\pi_l^*$, for playing air hockey. This is achieved by training within a simulated environment. The aim is to establish an expert policy, parameterized by $\theta$, that is adept at repelling the puck into the goal in a way that is replicable and understandable for human learners.

Training an expert policy for high-speed physical tasks like air hockey presents significant challenges due to numerous geometric, mechanical, and safety constraints. For example, in an air hockey game, the player must avoid actions that could damage the environment. It also has to operate within its physical capabilities, not exceeding its range of motion or velocity limits. Crucially, it should also limit its action space to mirror the range typical of human actions, ensuring that the policies can be later transferred to human learners. These constraints are not accounted for in the task's reward function, yet they are essential for successful policy training. In essence, the reward function does not adequately specify these important considerations. As a result, standard deep reinforcement learning methods are generally ineffective in producing suitable policies since they struggle to incorporate these vital physical constraints.

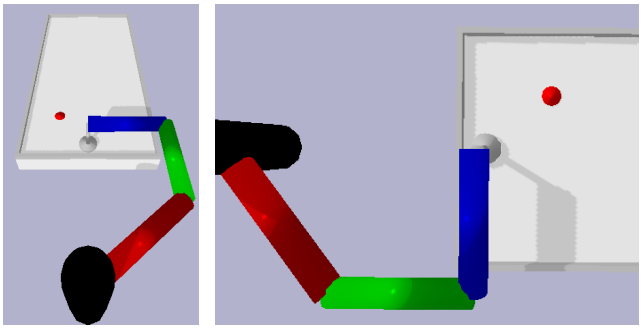


Fig. 5: Simulation environment based on PyBullet [59] used to train the expert policy. The puck is spawned randomly at the opposite side of the table and the 3-DOF robotic arm is trained to repel the puck into goal.

To effectively incorporate the constraints critical for policy training, we formulate the air hockey game as a Constrained Markov Decision Process (CMDP) and solve it using a dedicated CMDP solver. We build our CMDP upon the standard two-player air hockey game, which is the target task defined in Section III-B). Assuming agent 1 is the expert, we focus on training agent 1's policy to reactively repel the puck. To isolate the reactive policy training from the wider task of inferring the opponent's strategy in the sequential game, we segment the game into single-interaction episodes in which the agent only repels the puck once. A simulated environment emulating the real-world air hockey setup hosts a robotic arm with three degrees of freedom. The air hockey game is modeled as a CMDP for agent 1 with the following objective:

$$\max_{\theta} \mathbb{E}_{s_t,a_t,s_{t+1}} \left[ \sum_{t=0}^{\Gamma} \gamma^t R_1(s_t, a_t^1, s_{t+1}) \right] \quad \text{s.t.} \quad c(q_t^1) \leq 0. \tag{1}$$

Here, $q_t^1 \in Q^1$ represents the controllable joint state of the robotic arm, and $s_t = [q_t^1, s_t^e]$ comprises the state of the environment and the arm configuration, with $c(\cdot)$ mapping the state to constraint values. In the air hockey game, we consider two constraint types — position, and velocity — to ensure operational compliance with the arm's capabilities and task requirements. Agent 2's action and pose are omitted since the puck state is directly generated by the simulator without the need for a hitting action.

To solve the CMDP formulated above, we employ the Acting on the TAngent Space of the Constraint Manifold (ATACOM) algorithm with Proximal Policy Optimization [60], detailed in [13]. ATACOM adeptly manages various forms of constraints during policy learning and is compatible with model-free reinforcement learning algorithms, which are crucial given the missing dynamic models of the humans we aim to emulate. This method narrows exploration to relevant areas, thus accelerating the learning process. Our reward function encourages the robotic arm to repel the puck rapidly and hit it into the goal as fast as possible. Further elaboration on it is provided in the supplementary materials. After simulation

training, we gather the expert policy $\pi_\theta$, which will be used to discover sub-skills in the subsequent sections.

### B. Skill Decomposition

We extract the physical "skills" from the learned expert policies. We define our sub-skills through a pair of target task environment states, which encompasses the velocity and pose of the puck, following the approach of [14]. Specifically, this pair of skills includes the initial state $s_0^e$, which includes how the puck starts to move toward the learner, and the hitting state, $s_h^e$, which describes the state of the puck after it is struck.

For each policy trained using ATACOM, we collected trajectories by generating different initial states, $s_0^e$, but with a fixed position and recording how the trained policy would repel the puck, especially the hitting state $s_h^e$. We filter out those trajectories where the policy failed to repel the puck into the goal and only keep those successful trajectories. To extract discrete skills from the trajectory, we follow Campos et al. [14] to use VQ-VAE [57] to extract the skill. One may also use other clustering algorithms to achieve a similar purpose to replace VQ-VAE. Given $n$ pairs of states $\{(s_0^e, s_h^e)\}^n$, we deploy a VQ-VAE to learn the discrete skills.

The VQ-VAE comprises an encoder $\phi$ and a decoder $\psi$, it first passes the input through the encoder $\phi$, which outputs a continuous representation, $e = \phi((s_0^e, s_h^e))$. This representation is then quantized to a discrete latent variable $z_q$ through a nearest-neighbor lookup in the embedding space $\mathcal{E} = \{e_1, e_2, \ldots, e_K\}$, where $K$ is the number of embeddings, which are randomly initialized vector. The quantization process can be formally represented as follows:

$$z_q = e_k, \quad k = \arg\min_j \|z - e_j\|_2. \tag{2}$$

The decoder $\psi$ is then used to reconstruct the input from the discrete latent variable, i.e., $(\widehat{s_0^e}, \widehat{s_h^e}) = \psi(z_q)$. The loss function for VQ-VAE is composed of three terms: a reconstruction loss, a quantization loss, and a commitment loss. Reconstruction loss ensures that the decoded output closely matches the original input, quantization loss maintains the effectiveness of the embedding space, and commitment loss stabilizes the training by penalizing large changes in the encoder's output space. These can be formulated as:

$$\mathcal{L}_{\text{recon}} = \|s_0^e - \widehat{s_0^e}\|_2 + \|s_h^e - \widehat{s_h^e}\|_2, \tag{3}$$

$$\mathcal{L}_{\text{quant}} = \|\boldsymbol{sg}(z) - z_q\|_2, \tag{4}$$

$$\mathcal{L}_{\text{commit}} = \beta\|z - \boldsymbol{sg}(z_q)\|_2, \tag{5}$$

where $\boldsymbol{sg}$ denotes the stop-gradient operator, and $\beta$ is a hyperparameter that controls the weight of the commitment loss. Each skill is represented as a discrete latent variable in the embedding space of the VQ-VAE. By decoding these embeddings, we can recover the initial state and hitting state and use them to guide the human learner. As a result, the skill, $\omega$ is then represented by the pair of states $\omega = (s_0^e, s_h^e)$. The target now becomes for each $s_0^e$, we aim to train humans to repel the puck to the goal. Since the latent space is discrete, we are able to obtain a discrete number of sub-skills to teach. However, on the downside, to effectively train a VQ-VAE, the size of the latent space, $K$, must be relatively large to ensure convergence. In practice, we do not need all $K$ embeddings as some of them might be similar to each other. Therefore, in practice, after decoding the embedding into sub-skills, we merge similar sub-skills based on the state similarity and sample 15 sub-skills for teaching.

We prefer to use the puck's state after being struck by the learner as a basis for our skills over expert trajectory segments for two main reasons. Firstly, imitating exact trajectories in high-speed sports is challenging; the critical aspect is the trajectory's outcome or the resulting state of the puck. Secondly, allowing learners some flexibility rather than strict adherence to expert paths facilitates the utilization of individual strengths or preferences. This can be more beneficial for personalized coaching, enabling tailored training that aligns with a player's preference.

**Algorithm 1** Multi-Armed Bandit Teaching Algorithm with Softmax Exploration

**Require:** Sub-skill set $\Omega$, total training rounds $L$, observation window size $d$, training instance counter $J$, exclusion threshold, $\tau$.
Initialize $J(\omega) \leftarrow 0$ for each $\omega \in \Omega$.
Initialize $r(\omega) \leftarrow 0$ for each $\omega \in \Omega$.
$Z \leftarrow \text{Sample}(\Omega, n)$.
**for** $l = 1$ to $L$ **do**
  **if** $\Omega = \emptyset$ **then**
    **break**
  **end if**
  Compute $P(\omega) = \frac{\exp(r(\omega))}{\sum_{\omega \in Z} \exp(r(\omega))}$, $\forall \omega \in Z$.
  Sample sub-skill $\omega$ from $P(\omega)$.
  Update $r(\omega)$ post-teaching $\omega$ using Eq. 6.
  $J(\omega) \leftarrow J(\omega) + 1$.
  **if** $J(\omega) > \tau$ and $\sum_{j=J(\omega)-\tau}^{J(\omega)} \frac{C_j}{\tau} = 1$ **then**
    $\Omega \leftarrow \Omega \setminus \{\omega\}$.
    $Z \leftarrow (Z \setminus \{\omega\}) \cup \{\omega'\}$ for some $\omega' \in \Omega \setminus Z$.
  **end if**
**end for**=0

### C. Adaptive Teaching Sequence

After discovering the full set of sub-skills, we need to determine the sequence for our robot coach to teach them to a human to enhance learning efficiency. We define this sub-skills sequencing challenge as a Multi-Armed Bandit Problem. With the set of sub-skills acting as the action space for our robot coach, denoted by $\Omega = \{\omega_1, \omega_2, \ldots\}$, our goal is to select a sequence of teaching skills that maximizes the human learner's efficiency.

The vast number of physical skills discovered, coupled with the lack of initial information about students' proficiency levels, compounds the complexity of selecting an optimal teaching sequence. To tackle this, we adapt the "Zone of Proximal Development and Empirical Success" (ZPDES) algorithm [15], which simplifies the global optimization challenge into a dynamic local skill selection process. More importantly, the dynamic local skill selection is steered by a heuristic

function based on empirical success-guided reward. It contains two main components, the Zone of Proximal Development (ZPD) and Empirical Success.

The key idea of ZPD is to narrow down the skill selection candidates to a small subset of skills that learners are capable of mastering independently, conditioned on their current abilities. However, implementing standard ZPD requires a detailed hierarchy of skills, categorized by difficulty and prerequisites, which is challenging to establish for physical skills due to their nuanced difficulty levels and intricate interdependencies. As a workaround, we incorporate visual instruction as supplementary teaching signals, enabling learners to effectively acquire skills. Specifically, we select a random subset of discovered skills to form a "zone", $Z$. When teaching a skill $\omega = (s_0^e, s_h^e)$ from $Z$, we provide visual instruction on $s_h^e$ as an additional guide to the learner. This strategy effectively leverages the ZPD's efficiency while circumventing the complexities of mapping out a causal skill structure.

To efficiently select skills from the $Z$, we adopt the empirical success rate as a measure of learning progress from immediate feedback. We argue that skills leading to significant learning advancements should be prioritized, as they are likely to motivate learners more effectively [56]. Thus, our reward function for the MAB algorithm assesses the learning progress of each sub-skill as follows:

$$r(\omega) = \frac{2}{d}\left(\sum_{j=t-d/2}^{t} C_j - \sum_{j=t-d}^{t-d/2} C_j\right), \tag{6}$$

where $C_j$ equals 1 if the sub-skill $\omega$ is successfully executed in the $j$-th training instance, $d$ is the size of observation window, and $t$ is the total number of training instances of sub-skill $\omega$. This function compares the success rates of the latest $d/2$ instances with those of the previous $d/2$, serving as a practical estimate of the progress of the performance. Sub-skills that are either fully mastered or consistently unattainable yield zero reward. Given the reward, we then apply softmax exploration to choose the action. In practice, we set $d = 4$, and the size of $Z$ to 5.

In our system, we use the binary feedback to form a continuous assessment of the learner's behavior, similar to previous works [61, 12]. This provides enough information for the robot coach to track the learner's progress, while being easy for the robot coach to understand. Although a continuous measure, such as repelling speed, may provide more information, it would, in turn, require stronger interpretation capability on the part of the robot coach. Therefore, in this work we decided to use the binary score because it is simple and informative.

In addition, we set an exclusion threshold, $\tau$, to remove the sub-skill from the zone. If, after $t > \tau$ instances, the cumulative success rate for the last $\tau$ instances, $\sum_{j=t-\tau}^{t} \frac{C_j}{\tau} = 1$, shows consistent success, that sub-skill is excluded from the zone. A new sub-skill is then randomly selected for inclusion. Empirically, we set $\tau$ to 3. This sub-skill selection process is illustrated in Algorithm 1.

## V. Research Questions

Our experiments are designed to investigate the following three questions:

1) How does the robot coaching system affect human learning outcomes? We hypothesize that the discovered sub-skills and adaptive teaching would both leading to better human learning outcome in terms of performance.
2) How does the robot coaching system affect human learning experiences? We hypothesize that the discovered sub-skills and adaptive teaching will leads to a better learning experiences in terms of lower perceived workload, better usability and higher trust on the robot coach.
3) What are the important factors that affect human learning outcomes and experiences? We investigate other than the two components in Play-to-Coach, what are other factors may affect the human learning outcome and experiences.

## VI. Experiment Setup

### A. Metrics

To answer the research questions, we use the following metrics in our user study, including both objective and subjective measures.

label=*0:*

1) Success Rate Improvement: This is the main metric that we use to evaluate the efficiency of different teaching algorithms. In particular, when a human learner repels the puck and the puck falls into the goal region, the round is marked as successful; otherwise, it is marked as a failure round.
2) Human Subjective Metrics: These metrics are to measure human's perceived experience with the robot coach.

### B. Questionnaire

We ask the participants to answer the following questions before the experiment starts: label=*0:*

1) Demographic Information: We collect the ages and genders of the participants.
2) Relevant Sports Experience: We obtained the previous sports experience of the participants related to air hockey.

After the experiment, the participant is asked to answer the following questions: label=*0:*

1) NASA Task Load Index [62]: we use the NASA Task Load Index to collect human-perceived task load.
2) SUS Usability [63]: we use the SUS Usability form to understand the human-perceived usability of the system.
3) MDMT [64]: we use the MDMT form to assess the human perception of trust in the robot coach.
4) Customized Experience Questionnaire: we design a 5-point Likert scale questionnaire and open questions to understand human feedback.

### C. Variants of Play-to-Coach

To assess the design choice of Play-to-Coach and the important factors that affect human learning, we executed controlled experiments with three variants of Play-to-Coach: label=*0:*

1) *Random*: this variant uniformly samples hitting velocity from the hitting range.
2) *Partition*: this variant uniformly partitions the velocity range into 16 different discrete values and then applies the MAB algorithms for teaching. The human learner is not demonstrated how to repel the puck.
3) *Random Skill*: this variant uniformly samples the sub-skill for teaching. The human learner is shown how to repel the puck.

### D. Detailed Procedure

The experiment is structured into four distinct stages: preparation, trial, training, and evaluation. During the preparation stage, learners are provided with instructions about the task and are allowed to engage in five trial rounds of the game. In each round, there is one complete interaction between the robot coach and the human learner, where the robot coach strikes the puck and the human learner attempts to repel it.

During the trial stage, the human learner engages in 30 rounds of play with the robot. The objective of this stage is to gather data on the learner's initial performance, which serves as a baseline to measure improvement post-training. The initial assessment is used only to calculate the improvement made by the human learner and is not used in the method. In these rounds, the robot employs a random policy, where it selects a velocity uniformly from a given range for each puck strike.

In the training stage, the robot would play according to certain teaching strategies, as we illustrate in the variant's descriptions. Human learners are trained for 100 rounds of the game. We allow the human learner to rest for 5 minutes between different stages and 2 minutes during the training stage. In the evaluation stage, the robot would deploy a random policy - the robot would sample a velocity in the given range uniformly. It lasts for 50 rounds to evaluate human learning outcomes compared with the trial stage. In total, one experiment takes 1 hour and 10 minutes to finish including filling in the survey. To remove the influence of prior experience of the game, participants are required to use the non-dominant hand to play the game.

We recruited 30 participants from a university campus to carry out the experiments. University IRB exemption is acquired for the experiment. Due to significant fatigue during the experiment, we removed the data from two players from the analysis, resulting in 28 participants with ages ranging from 18 to 32 years old ($M$ = 25.64, $SD$ = 3.82, 7 females) in total and uniformly assigned them to 4 groups for experiments between subjects. All participants involved in the study either possess a bachelor's degree or are currently pursuing one. We interviewed the participants about their prior experience with air hockey; only one of the participants had previous experience with air hockey before. Therefore, the human subjects involved in the experiment are mainly novices. Each participant received $10 for finishing the experiment. In addition, to motivate the learning of the skill, participants with a top-3 success rate among all participants during the evaluation stage will receive an additional $20 as a bonus.

## VII. Experiment Results

The analysis was mainly performed using a 2x2 factorial ANOVA to determine the influence of skill decomposition and MAB-based adaptive teaching on human learning. The result of objective metrics and subjective metrics are shown in Fig. 6 and Fig. 7 respectively.

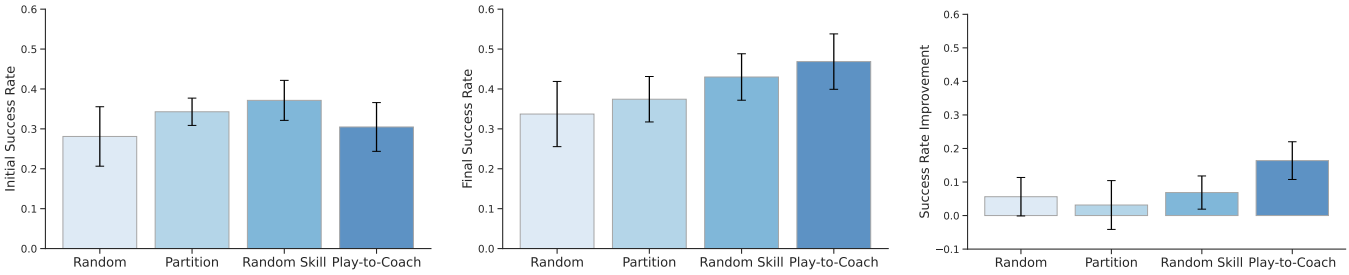
### A. Objective Analysis

*1) How do discovered sub-skills help with human learning outcomes?:* We compare the improvement of the success rate in different groups before and after training. The statistical analysis showed that the skill decomposition leads to significantly better success rate improvement ($F_{1,24}$ = 4.858, $p$ = 0.037, $\eta^2$ = 0.146). Subsequent post hoc examinations employing the Tukey HSD test revealed that these discovered sub-skills and visual instruction significantly aid human learners in improving their abilities efficiently ($p$ = 0.037, Cohen's d = 0.833). This support our hypothesis that the discovered sub-skill is helpful to improve human learning outcome.

During post-experiment interviews, 12 of 14 participants in the group who participated in skill discovery reported finding the demonstrated visual instruction of the sub-skill beneficial in their learning process. Particularly, that saves participants' effort in exploration, allowing them to concentrate more on low-level motor control aspects. Additionally, participants were asked about the sub-skills they considered most useful. The majority highlighted a specific sub-skill that involves using the table boundary to create a reflective angle, allowing the puck to be directed toward the goal.

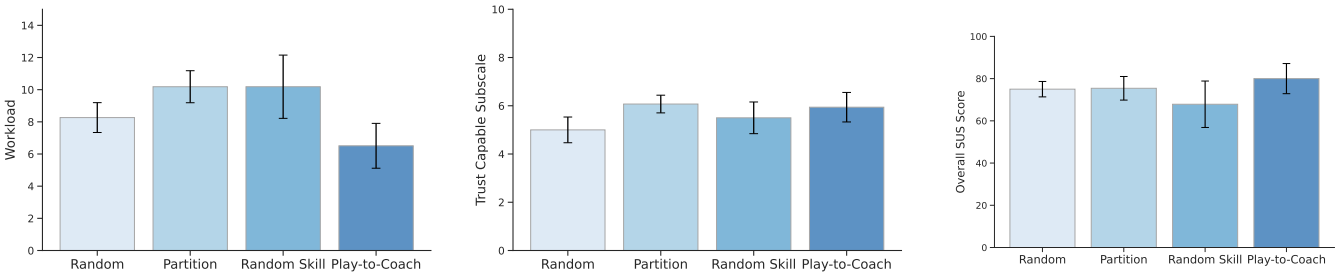
*2) How does the derived teaching sequence help with human learning outcomes?:* The statistical analysis showed that neither the use of the MAB algorithm ($F_{1,24}$ = 1.151, $p$ = 0.294, $\eta^2$ = 0.035) nor the interaction effect ($F_{1,24}$ = 3.338, $p$ = 0.080, $\eta^2$ = 0.100) appear to significantly improve human learning outcomes, which rejects our hypothesis that the adaptive teaching help improve human learning outcome.

We hypothesize that the inherent difficulty of physical skills makes the MAB problem in our case more demanding than the other educational domain. Therefore, we perform analysis on the partition group and Play-to-Coach group. Although the sub-skill to be taught is different, the number of sub-skills humans are considered to have mastered using the MAB algorithm differs significantly in these two groups Table I. The number of skills mastered by participants in the partition group is significantly less than that of the Play-to-Coach group. We hypothesize that randomly partitioning the initial velocity into partitions would lead to some hard cases that are hard to master without explicit teaching or scaffolding. Then, purely relying on the learner's exploration of the solution might be hard and inefficient [15]. This results in the situation where the




*a*) Success Rate Before Training *b*) Success Rate After Training *c*) Success Rate Improvement

Fig. 6: Objective measures.




*a*) NASA TLX Workload *b*) MDMT Trust Capability Subscale *c*) SUS Usability Score

Fig. 7: Subjective measures.

TABLE I: Skill mastered by human participants in groups using the MAB algorithm. The human learner is considered to have mastered the sub-skill if they have successfully repelled the puck into the goal three consecutive times when trained with that sub-skill.

| Group | Partition | Play-to-Coach |
|---|---|---|
| #Skill Mastered | 5.00 $\pm$ 1.51 | 8.00 $\pm$ 2.51 |

robot coach would keep teaching the same set of sub-skills and fail to cover the others, thus, worsening the human learning outcome.

### B. Subjective Analysis

*1) Do the teaching methods help reduce human workload and increase usability?:* Now we shift our focus to human-perceived training experience, we first assess human-perceived workload. We aggregated the responses to the NASA TLX form to obtain a single score for each participant averaging the score from six categories. We again run a 2x2 factorial ANOVA to analyze the data. Given the analysis result, the interaction effect shows significance in reducing the human perceived workload ($F_{1,24}$ = 13.430, $p$ = 0.001, $\eta^2$ = 0.335). Post hoc comparisons using the Tukey HSD test revealed a significantly higher perceived workload in the random skill group than Play-to-Coach group ($p$ = 0.012, Cohen's d = 1.82), highlighting the importance of focused and adaptive coaching in the human learning experience. Furthermore, it was observed that the sub-skill involving the puck being hit to the side was taught after the sub-skill involving the puck being hit to the center in the generated adaptive curriculum. This may contribute to the reduction in perceived workload.

However, despite the human perceived workload is reduced, when considering the SUS usability scores across different groups, we did not observe significant difference among groups. This outcome suggests that, despite the varying teaching methods, the overall usability of the system from the user's perspective remains consistent.

*2) How would the teaching system affect human perceived trust in the system?:* We used the MDMT questionnaire to collect human trust in the robot coach and only calculated the capability subscale since there is no social interaction between the human subject and the robot coach. We again run 2x2 factorial ANOVA to analyze the data. We found that the MAB algorithm improves human trust in the robot's capability. The results show the MAB algorithm which enables focused training leads to higher human trust ($F_{1,24}$ = 6.149, $p$ = 0.021, $\eta^2$ = 0.195) and post-hoc test using Tuky HSD test shows the significance ($p$ = 0.021, Cohen's d = 0.937). This is also partially supported by our survey. In the customized questionnaire, we asked human participants in groups not using MAB algorithms about any improvement that can be made to the current system; 6 out of 14 mentioned more focused training. We found that such focused training can be easily perceived by human participants without explicitly telling them, thus leading to a better image of a coach in the human mind.

Although adaptive teaching increases the human learner's confidence in the system, we observed that the human learner may not use the learned skill during evaluation if the evaluation case is similar to the training case. Therefore, for the Random Skill and Play-to-Coach groups, we conducted post-questionnaires asking participants why they did not use the learned skill during evaluation, if there was any reason.

Participants' feedback revealed two main reasons for their reluctance to use the learned skills: 1) lack of confidence in mastering the skill ( *"I am not sure I can hit it in"*), and 2) mistrust of the robot's visual instructions ( *"I think hitting in other ways is easier"*). Such a trust problem is exacerbated in physical training tasks because there's often no default or standardized way to perform certain actions, as there is in some other training domains.

*Trust in the Instruction:* Trust in the instruction is closely related to the robot's ability to clearly explain and justify its teaching methods. We found that 7 of 14 participants in the group using skill decomposition desired more comprehensive explanations of the visual instruction when further asked for suggestions for the system. This echoes the importance of transparency in AI systems as suggested by [65]. The need for clearer justifications is indicative of a gap in the robot's communicative competence, a crucial factor for establishing trust in HRI as per the Technology Acceptance Model (TAM) [66]. The TAM suggests that perceived usefulness and ease of use significantly influence the acceptance of technology, which in this case, relates to the robot's credibility.

In addition, the adaptivity of teaching robots, particularly in skill discovery, is crucial in the distrust problem. Current methods, like MAB algorithms, do not fully accommodate individual learner differences such as skill levels or personal preferences. This shortcoming can result in sub-skills being perceived as not fitting their preference according to the open question in our questionnaire.

*Trust in the Outcome:* The concept of outcome trust in robot-led physical training intertwines with the learner's belief in their ability to apply the skills effectively. Our findings align with Bandura's Self-Efficacy Theory [67], which posits that self-confidence in skill execution is critical for learning. In our task, the effect is amplified since the task is highly dynamic. This correlation is evident in participants' reluctance to employ skills they were not confident about. Hence, enhancing feedback mechanisms in line with suggestions from [38] might bolster self-efficacy and trust in training outcomes.

### C. Core Findings

In conclusion, we show that sub-skills with visual instructions from the learned policy effectively enhance learning outcomes, yet the adaptive teaching algorithm alone is insufficient for significant improvement in complex physical skills. Its effectiveness is primarily seen in enhancing the human learning experience, notably by reducing perceived workload together with skill decomposition and fostering trust through focused training.

These findings underscore the critical role of human trust and the adaptability of sub-skills in the realm of physical robot teaching. Trust in the outcomes and instructions from the robot coach greatly influences learner behavior, suggesting the need for improved communication between the robot coach and the human learner. Furthermore, the current approach to skill decomposition overlooks individual human preferences, leading to a lack of adaptability and varied perceptions of the utility of sub-skills and guidance. This highlights a clear need for more personalized and adaptable approaches in robotic teaching systems to better cater to individual learner needs and preferences.

## VIII. Limitations

One area for improvement is conducting a larger-scale human subject study to further validate and support the findings presented in this work. The other major limitation of our work is the focus solely on the short-term effects of the teaching results. This scope restricts our ability to understand the long-term retention and application of skills learned through the robot coaching system. It is important to acknowledge that the effectiveness of teaching methods, especially in complex physical tasks, is often more accurately assessed over an extended period. This prolonged evaluation could provide valuable insights into how well skills are retained, the long-term impact of reduced workload and enhanced trust, and the effectiveness of adaptive teaching strategies over time. Additionally, observing long-term effects could reveal more about the evolving relationship and trust dynamics between the human learner and the robot coach, as well as potential shifts in learners' preferences and adaptability to the taught skills. Future research could aim to address this limitation by conducting long-term studies, thereby offering a more comprehensive understanding of the impact and efficacy of robotic coaching systems in physical skill learning.

## IX. Discussion on Generalization

In Play-to-Coach, we focus on one task – air hockey. While we have found interesting results that can be generalized to several different scenarios, generalization to additional tasks will be an important step in demonstrating true generalization. In this section, we discuss how Play-to-Coach can be generalized to additional tasks. To extend the teaching system to more complex, high-dimensional physics problems, three key factors are crucial: skill decomposition, skill structure discovery, and multimodal interaction.

First, a more general approach to decomposing skills is crucial for further scaling the system to more complex tasks. Decomposing complex skills into more manageable sub-skills is a common strategy for simplifying tasks. Existing frameworks often represent sub-skills as trajectory segments, a method that simplifies skills into sequential steps. However, this approach may not always be optimal because it assumes consistent, predictable skill execution, which doesn't fully account for the dynamic and varied nature of human or system interactions in complex environments. Interactions with other participants can drastically alter the context and execution of a task. An alternative is a goal-oriented representation, where sub-skills are defined by their goals rather than specific actions. This strategy enhances flexibility and adaptability, allowing for different methods to achieve the same goals and accommodating differences in abilities, environmental factors, or task demands. In our system, we use the resulting puck state as a goal to represent sub-skills. Identifying such a

goal-oriented representation is crucial for generalizing our approach.

Second, in the context of complex, high-dimensional physical problems, it is not enough to identify the sub-skills; the structure of these sub-skills must also be elucidated. Mathematics education benefits from a natural decomposition into different concepts, which facilitates the decomposition of complex knowledge. However, without understanding basic concepts such as addition and multiplication, the acquisition of more complex knowledge becomes a challenge for the learner. It is therefore essential to understand the structure of knowledge, for example in the form of prerequisites, if one is to complete complex tasks that involve a hierarchical learning process. One straightforward solution to this is to use the learned expert policy to rank the sub-skill to be learned according to its own success rate. However, this may not accurately describe the difficulty a human learner may perceive due to different learning preferences. One solution is to use the initial assessment of a large group of participants to learn the structure based on human perceived difficulty.

Finally, we must expand the interaction modes. Our studies with human subjects revealed a clear preference for explanations of the robot coach's behavior. This is particularly important for more complex problems, where physical interaction alone may not be sufficient or the most effective means of conveying the information for learners to grasp the underlying concepts. It is clear that supplementary modalities, such as language or visual aids, can provide information that is difficult to convey through physical interaction alone. This is in line with the findings of [38].

## X. Conclusion

In conclusion, we introduce Play-to-Coach, an autonomous robot coaching system that demonstrates its effectiveness in enhancing the learning of physical skills. Our findings highlight that skill discovery from learned policies enhances learning outcomes, while adaptive teaching notably improves the overall learning experience. Furthermore, our research underscores the importance of fostering human trust and adaptability in interactions with robot coaching systems. Future work could focus on refining these systems for a broader range of physical skills, exploring long-term learning impacts, and enhancing personalization to cater to diverse learning styles and preferences. This will further solidify the role of autonomous robots in aiding humans to learn physical skills.

## References


[1] D. P. Losey, C. G. McDonald, E. Battaglia, and M. K. O'Malley, "A review of intent detection, arbitration, and communication aspects of shared control for physical human–robot interaction," *Applied Mechanics Reviews*, 2018.

[2] M. Arduengo, A. Colomé, J. Borràs, L. Sentis, and C. Torras, "Task-adaptive robot learning from demonstration with gaussian process models under replication," *IEEE Robotics and Automation Letters*, 2021.

[3] Y. Zhou, J. Gao, and T. Asfour, "Learning via-point movement primitives with inter-and extrapolation capabilities," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[4] Y. Li, K. P. Tee, W. L. Chan, R. Yan, Y. Chua, and D. K. Limbu, "Continuous role adaptation for human–robot shared control," *IEEE Transactions on Robotics*, 2015.

[5] A. U. Pehlivan, D. P. Losey, and M. K. O'Malley, "Minimal assist-as-needed controller for upper limb robotic rehabilitation," *IEEE Transactions on Robotics*, 2015.

[6] D. P. Losey, H. J. Jeon, M. Li, K. Srinivasan, A. Mandlekar, A. Garg, J. Bohg, and D. Sadigh, "Learning latent actions to control assistive robots," *Autonomous robots*, 2022.

[7] C. Yu, Y. Xu, L. Li, and D. Hsu, "Coach: Cooperative robot teaching," in *Conference on Robot Learning*, 2023.

[8] M. Srivastava, E. Biyik, S. Mirchandani, N. Goodman, and D. Sadigh, "Assistive teaching of motor control tasks to humans," in *Advances in Neural Information Processing Systems*, 2022.

[9] J. Gonzalez-Brenes, Y. Huang, and P. Brusilovsky, "General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge," in *Educational Data Mining*, 2014.

[10] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein, "Deep knowledge tracing," in *Advances in Neural Information Processing Systems*, 2015.

[11] X. Xiong, S. Zhao, E. V. Inwegen, and J. E. Beck, "Going deeper with deep knowledge tracing," in *Educational Data Mining*, 2016.

[12] T. Schodde, K. Bergmann, and S. Kopp, "Adaptive robot language tutoring based on bayesian knowledge tracing and predictive decision-making," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2017.

[13] P. Liu, D. Tateo, H. B. Ammar, and J. Peters, "Robot reinforcement learning on the constraint manifold," in *Conference on Robot Learning*, 2022.

[14] V. Campos, A. R. Trott, C. Xiong, R. Socher, X. G. i Nieto, and J. Torres, "Explore, discover and learn: Unsupervised discovery of state-covering skills," in *International Conference on Machine Learning*, 2020.

[15] B. Clement, D. Roy, P.-Y. Oudeyer, and M. Lopes, "Multi-armed bandits for intelligent tutoring systems," *Journal of Educational Data Mining*, 2015.

[16] S. S. Gu, E. Holly, T. P. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," *International Conference on Robotics and Automation (ICRA)*, 2016.

[17] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," in *Conference on Robot Learning*, 2022.

[18] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *Conference on Robot Learning*, 2021.

[19] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Mart'in-Mart'in, "What matters in learning from offline human demonstrations for robot manipulation," in *Conference on Robot Learning*, 2021.

[20] H.-T. L. Chiang, A. Faust, M. Fiser, and A. Francis, "Learning navigation behaviors end-to-end with autorl," *IEEE Robotics and Automation Letters*, 2018.

[21] M. Carroll, R. Shah, M. K. Ho, T. L. Griffiths, S. A. Seshia, P. Abbeel, and A. D. Dragan, "On the utility of learning about humans for human-ai coordination," in *Advances in Neural Information Processing Systems*, 2019.

[22] S. Ross and D. Bagnell, "Efficient reductions for imitation learning," in *International Conference on Artificial Intelligence and Statistics*, 2010.

[23] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning*, 2022.

[24] A. R. Mahmood, D. Korenkevych, G. Vasan, W. Ma, and J. Bergstra, "Benchmarking reinforcement learning algorithms on real-world robots," in *Conference on robot learning*, 2018.

[25] J. Ibarz, J. Tan, C. Finn, M. Kalakrishnan, P. Pastor, and S. Levine, "How to train your robot with deep reinforcement learning: lessons we have learned," *The International Journal of Robotics Research*, 2021.

[26] D. Hadfield-Menell, S. Milli, P. Abbeel, S. J. Russell, and A. D. Dragan, "Inverse reward design," *ArXiv*, 2017.

[27] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual review of control, robotics, and autonomous systems*, 2020.

[28] E. K. Hedlund, M. Johnson, and M. C. Gombolay, "The efects of a robot's performance on human teachers for learning from demonstration tasks," *2021 16th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2021.

[29] J. Psotka, L. D. Massey, and S. A. Mutter, *Intelligent tutoring systems: Lessons learned*, 1988.

[30] M. S. Lee, H. Admoni, and R. Simmons, "Machine teaching for human inverse reinforcement learning," *Frontiers in Robotics and AI*, 2021.

[31] X. Zhu, "Machine teaching: An inverse problem to machine learning and an approach toward optimal education," *AAAI Conference on Artificial Intelligence*, 2015.

[32] J. Liu, X. Zhu, and H. Ohannessian, "The teaching dimension of linear learners," in *International Conference on Machine Learning*, 2016.

[33] S. Mei and X. Zhu, "Using machine teaching to identify optimal training-set attacks on machine learners," in *AAAI Conference on Artificial Intelligence*, 2015.

[34] F. Khan, B. Mutlu, and J. Zhu, "How do humans teach: On curriculum learning and teaching dimension," in *Advances in Neural Information Processing Systems*, 2011.

[35] H. Kose-Bagci and R. Yorganci, "Tale of a robot: Humanoid robot assisted sign language tutoring," *International Conference on Humanoid Robots*, 2011.

[36] H. Kose-Bagci, R. Yorganci, E. H. Algan, and D. S. Syrdal, "Evaluation of the robot assisted sign language tutoring using video-based studies," *International Journal of Social Robotics*, 2012.

[37] N. Randall, "A survey of robot-assisted language learning (rall)," *ACM Transactions on Human-Robot Interaction (THRI)*, 2019.

[38] M. Srivastava, N. Goodman, and D. Sadigh, "Generating language corrections for teaching physical control tasks," in *40th International Conference on Machine Learning (ICML)*, 2023.

[39] R. Tian, M. Tomizuka, A. D. Dragan, and A. V. Bajcsy, "Towards modeling and influencing the dynamics of human learning," *International Conference on Human-Robot Interaction(HRI)*, 2023.

[40] A. D. Dragan and S. S. Srinivasa, "A policy-blending formalism for shared control," *International Journal of Robotics Research*, 2013.

[41] S. Reddy, A. D. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," in *Robotics: Science and Systems*, 2018.

[42] J. Leike, D. Krueger, T. Everitt, M. Martic, V. Maini, and S. Legg, "Scalable agent alignment via reward modeling: a research direction," *CoRR*, 2018.

[43] H. J. Jeon, S. Milli, and A. Dragan, "Reward-rational (implicit) choice: A unifying formalism for reward learning," in *Advances in Neural Information Processing Systems*, 2020.

[44] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems*, 2017.

[45] A. Fern, S. Natarajan, K. Judah, and P. Tadepalli, "A decision-theoretic model of assistance," *Journal of Artificial Intelligence Research*, 2014.

[46] R. Shah, P. Freire, N. Alex, R. Freedman, D. Krasheninnikov, L. Chan, M. D. Dennis, P. Abbeel, A. Dragan, and S. Russell, "Benefits of assistance over reward learning," 2021.

[47] O. Macindoe, L. Pack Kaelbling, and T. Lozano-Pérez, "Pomcop: Belief space planning for sidekicks in cooperative games," *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2021.

[48] B. J. Grosz and S. Kraus, "Collaborative plans for complex group action," *Artificial Intelligence*, 1996.

[49] S. Nikolaidis, A. Kuznetsov, D. Hsu, and S. Srinivasa, "Formalizing human-robot mutual adaptation via a bounded memory based model," in *ACM/IEEE International Conference on Human Robot Interaction*, 2016.

[50] S. Nikolaidis, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in collaborative tasks: Models and experiments," *International Journal of Robotics Research*, 2017.

[51] S. Nikolaidis, Y. X. Zhu, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in shared autonomy," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2017.
[52] A. Xie, D. P. Losey, R. Tolsma, C. Finn, and D. Sadigh, "Learning latent representations to influence multi-agent interaction," *Annual Conference on Robot Learning*, 2020.
[53] W. Z. Wang, A. Shih, A. Xie, and D. Sadigh, "Influencing towards stable multi-agent interactions," *Conference on Robot Learning*, 2021.
[54] A. N. Rafferty, E. Brunskill, T. L. Griffiths, and P. Shafto, "Faster teaching by pomdp planning," in *Artificial Intelligence in Education*, 2011.
[55] J. P. González-Brenes and J. Mostow, "What and when do students learn? fully data-driven joint estimation of cognitive and student models," in *Educational Data Mining*, 2013.
[56] J. Gottlieb, P.-Y. Oudeyer, M. Lopes, and A. Baranes, "Information-seeking, curiosity, and attention: computational and neural mechanisms," *Trends in Cognitive Sciences*, 2013.
[57] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *ArXiv*, 2017.
[58] P. Liu, D. Tateo, H. Bou-Ammar, and J. Peters, "Efficient and reactive planning for high speed robot air hockey," in *International Conference on Intelligent Robots and Systems (IROS)*, 2021.
[59] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016–2021.
[60] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms." *CoRR*, 2017.
[61] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham, "Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation," in *Educational Data Mining*, 2016.
[62] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," *Advances in psychology*, 1988.
[63] J. B. Brooke, "Sus: A 'quick and dirty' usability scale," 1996.
[64] D. Ullman and B. F. Malle, "Measuring gains and losses in human-robot trust: Evidence for differentiable components of trust," *International Conference on Human-Robot Interaction (HRI)*, 2019.
[65] Z. C. Lipton, "The mythos of model interpretability," *Communications of the ACM*, 2016.
[66] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Q.*, 1989.
[67] A. Bandura, "Self-efficacy: toward a unifying theory of behavioral change." *Psychological review*, 1977.