Received: 11 April 2024 | Revised: 17 June 2024 | Accepted: 19 June 2024

DOI: 10.1002/cae.22780

RESEARCH ARTICLE

WILEY

Teaching approach for deep reinforcement learning of robotic strategies

Janez Podobnik | **Ana Udir** | **Marko Munih** | **Matjaž Mihelj**

Laboratory of Robotics, University of Ljubljana, Faculty of Electrical Engineering, Ljubljana, Slovenia

Correspondence
Janez Podobnik, Laboratory of Robotics, Faculty of Electrical Engineering, University of Ljubljana, Ljubljana, Slovenia.
Email: janez.podobnik@fe.uni-lj.si

Present address
Janez Podobnik, Laboratory of Robotics, Faculty of Electrical Engineering, University of Ljubljana, Ljubljana, Slovenia.

Funding information
Javna Agencija za Raziskovalno Dejavnost RS

Abstract

This paper presents the development of a teaching approach for Reinforcement Learning (RL) for students at the Faculty of Electrical Engineering, University of Ljubljana. The approach is designed to introduce students to the basic concepts, approaches, and algorithms of RL through examples and experiments in both simulation environments and on a real robot. The approach includes practical programs written in Python and presents various RL algorithms. The Q-learning algorithm is introduced and a deep Q network is implemented to introduce the use of neural networks in deep RL. The software is user-friendly and allows easy modification of learning parameters, reward functions, and algorithms. The approach was tested successfully on a Franka Emika Panda robot, where the robot manipulator learned to move to a randomly generated target position, shoot a real ball into the goal, and push various objects into target position. The goal of the presented teaching approach is to serve as a study aid for future generations of students of robotics to help them better understand the basic concepts of RL and apply them to a wide variety of problems.

KEYWORDS

augmented reality, reinforcement learning, robotics education, sensors

1 | INTRODUCTION

Artificial Intelligence (AI) has become more accessible through open-source software, contributing to the significant growth and success of Reinforcement Learning (RL) [5, 34]. RL enables autonomous agents to learn through trial and error, making it increasingly prevalent in various industries [15]. RL is a vital component of AI-enhanced robotic systems and an indispensable skill for future robotics experts, particularly in the era of the 4th and 5th Industrial Revolution [26, 36]. Therefore, it is crucial for robotics students to acquire a foundational understanding of RL and other AI technologies [11, 13].

RL is widely used across various domains such as self-driving cars, healthcare, recommendations, news, advertisements, and gaming. In the field of robotics, RL holds particular relevance [21, 27, 30]. Recognizing its broad applicability and complexity, the Laboratory of Robotics at the Faculty of Electrical Engineering aims to introduce students to RL, a rapidly advancing area of artificial intelligence. To effectively convey the fundamental concepts and principles of RL, the use of experiments and examples is essential [6]. To showcase the algorithms, we require diverse problem environments of varying difficulty levels. Our objective is to develop a comprehensive teaching approach that seamlessly integrates simulations and real-world scenarios, combining problem-solving with RL algorithms. This approach will bridge the gap between theoretical knowledge and practical application in robotics, serving as a valuable study resource to comprehend the theoretical and practical intricacies and breadth of RL.

Educational robotics has been extensively studied and proven effective in enhancing student engagement, particularly in primary and secondary education [14, 29, 37, 41]. This is reflected in the predominant focus of published studies that focus on the use of low-cost mobile robots in educational settings [10, 25]. However, there is a lack of literature documenting the use of robots in higher education, specifically in teaching RL with research and industrial robots [29, 31, 42]. Suenaga et al. [33] presented a web-based system for studying mobile robot navigation using deep RL, but it only allows parameter setting without providing students the opportunity to learn algorithms or apply RL to diverse problems. Toivonen et al. [35] utilized Lego Mindstorms EV3 robots and the Open Learning Environment for Artificial Intelligence (OLE-AI) to teach RL and artificial neural networks to computer science students. Martínez-Tenor et al. [19] designed interactive sessions for RL application in the "Cognitive Robotics" master's degree program using Lego Mindstorms robots, employing a mixed instructivist-constructivist approach that enables students to grasp theory and shape their own educational experience.

There may be several reasons for the scarcity of published studies documenting the use of research and industrial robots in teaching RL in higher education, including technical challenges, cost, limited access, and complexity [6, 16, 38]. Educators may not have the technical expertise and specialized knowledge needed to use these robots in the classroom [8, 29], and institutions may have difficulty funding and maintaining them [4, 19]. Limited access to research and industrial robots also hinders students from gaining practical experience. Moreover, given the complexity of RL, simulation environments and educational robots may offer more practical, accessible, and cost-effective alternatives [39]. Overcoming these challenges and finding innovative solutions will be critical in advancing the use of research and industrial robots in RL teaching.

The application of robots in higher education to teach RL is crucial for several reasons [17]. First, it allows students to bridge the gap between theoretical concepts and practical applications [31], enabling them to develop a deep understanding of RL principles in real-world contexts. Second, as the demand for professionals with expertise in RL continues to grow, incorporating robots into higher education curricula helps prepare students for the challenges and opportunities they will face in their future careers. Third, working with robots presents unique problem-solving challenges that foster the development of critical thinking and creativity. Despite these compelling reasons, there is limited literature on using robots in higher education to teach RL, highlighting the need for further exploration and documentation of these practices.

This paper presents the integration of RL into a robot control course, emphasizing its importance for robotics students. RL is often absent from mandatory robotics courses, which highlights the need for its inclusion. We have developed a teaching approach and tools to enhance students' understanding of RL's fundamental concepts. Our approach enables students to test algorithms using real-world examples, observe their effects, and apply RL to diverse problems. The study program's content is designed to facilitate the explanation of RL principles, providing lecturers and students with valuable teaching resources. The novelty of this work lies in its focus on teaching RL to students using both simulation environments and real robots, offering students a hands-on learning experience. This approach deepens their understanding of RL and enhances their ability to apply it in practical situations. We believe that our comprehensive and coherent teaching approach, combining these elements, is a unique and valuable contribution to the field.

The paper explores three topics:

1. Impact of the comprehensive course on Reinforcement Learning in robotics on students' understanding of RL concepts and their ability to apply RL techniques to solve complex robotic problems.
2. Impact of including a physical robot manipulator in the RL education of students compared to virtual simulations alone.
3. The factors that contribute to the effectiveness of the course in teaching of RL in robotics.

2 | MATERIALS AND METHODS

2.1 | A short introduction to RL

RL differs from supervised and unsupervised learning in that it does not require prior information for initial action [5, 34]. It gradually acquires the information needed for learning as it progresses. In RL, the agent learns to behave in the environment by performing different actions and evaluating their effects and appropriateness. The agent is not explicitly told what to do at a given

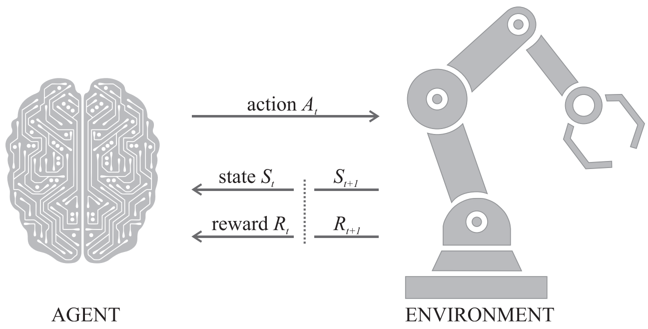

FIGURE 1 Principle diagram of Reinforcement Learning.

moment in time but learns through repetition by performing actions and evaluating their success with the rewards it receives. The action is chosen by the agent according to the strategy (policy) it follows. A policy is a function that defines the choice of action given the current environment and knowledge (see Figure 1). The better the action chosen, the higher the reward. The goal of the agent is to maximize the accumulated reward by performing actions.

RL encompasses various algorithms [34]. Here, we provide an overview of the learning progression in programming and utilizing RL algorithms. Students start with implementing the Q-learning algorithm and then advance to employing deep learning methods for RL.

One of the most fundamental RL algorithms is Q-learning [34], which takes its name from the term "quality" (Q), as we define the quality of an action within a given state. It is a model-free, off-policy algorithm that employs the temporal-difference (TD) method. The algorithm learns the Q-values of state-action pairs, approximating the optimal function of the Q-values of state-action pairs, regardless of the strategy employed. In its simplest form, Q-learning is represented by a look-up table, with each cell corresponding to a value of a state-action pair. The algorithm is defined as follows:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha [R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)], \quad (1)$$

where $Q(S_t, A_t)$ denotes the value of the state-action pair being updated, α denotes the learning rate of the algorithm, R_{t+1} represents the reward obtained upon transitioning from state S_t to state S_{t+1} through action A_t, and a discount factor γ is the decrementing parameter of the reward value. The expression $maxQ(S_{t+1}, a)$ denotes the maximum value of the state-action pair that is attainable from state S_{t+1}. The agent takes actions based on an exploration strategy, such as ε-greedy, that balances between exploration and exploitation. The Q-values are updated after each action, and the process continues until the Q-values converge or a maximum number of iterations is reached. Once the Q-values are learned, the agent can employ greedy strategy to select actions that maximize the expected reward in a given state. The procedure is described in Algorithm 1.

Algorithm 1 Q-learning

Initialize arbitrary values $Q(S, A)$ for every $S \in \mathcal{S}, A \in \mathcal{A}$. Define learning rate $\alpha \in (0, 1]$, number of episodes N, maximum number of steps per episode $step_{max}$, discount factor γ, and parameter $\varepsilon \in [0, 1]$.
while *episode*!= N **do**
  S = initial state
  step= 0
  while S!= *final state* **and** *step*!= $step_{max}$ **do**
    Based on strategy (e.g. ε-greedy) in state S, select action A.
    Save obtained reward R and new state S'
    Update values of $Q(S, A)$ and S:
    $Q(S, A) \leftarrow Q(S, A) + \alpha [R + \gamma \max_a Q(S', a) - Q(S, A)]$
    $S \leftarrow S'$
    step += 1
  end while
  episode += 1
end while

Deep RL methods employ neural networks to approximate complex value functions or policies. These approaches are very useful for high-dimensional and continuous state spaces. The progress in deep RL has led to two main approaches: Q-Learning and Policy Optimization. Q-Learning approximates the Q-function and has been fundamental in deep RL. Deep Q-Networks (DQN) introduced deep neural networks to approximate Q-values, facilitating efficient handling of large state spaces.

On the other hand, Policy Optimization methods learn the optimal policy directly without explicitly estimating Q-values. Algorithms like Proximal Policy Optimization (PPO) and Trust Region Policy Optimization (TRPO) iteratively update policy parameters using gradient-based techniques while ensuring policy improvements.

Each approach possesses distinct advantages and disadvantages. Q-function approximation methods are generally more sample-efficient but can be challenging to train in high-dimensional state spaces. Policy gradient methods are typically less sample-efficient but exhibit greater robustness to high-dimensional state spaces.

2.2 | Course syllabus

In this section we will provide a brief overview of the course syllabus.

1. Objectives:
   - Provide students with a comprehensive understanding of the theoretical foundations and concepts of RL in the context of robotics.
   - Enable students to apply RL techniques to solve real-world robotic problems and develop the necessary skills to implement and experiment with Reinforcement Learning algorithms.
   - Foster problem-solving abilities through hands-on practical exercises.
2. Competencies that students gain:
   - Knowledge and understanding of the fundamental principles and concepts of Reinforcement Learning.
   - Implementing and experimenting with RL algorithms using software libraries and simulation environments.
   - Capability to transfer RL models trained in simulations to physical robot manipulators.
3. Intended learning outcomes:
   - Demonstrate a solid understanding of RL concepts, algorithms, and techniques in the context of robotics, and apply them effectively to solve robotic problems.
   - Develop competence in adapting RL techniques to specific scenarios, demonstrating the ability to select and fine-tune appropriate algorithms for different robotic applications.
   - Design and implement RL models using software libraries and simulation environments, and successfully transfer and deploy these models on physical robot manipulators to demonstrate their performance in real-world scenarios.
4. Learning and teaching methods:
   - Lectures (Cognitivism): Provide theoretical knowledge and explanations of RL concepts, algorithms, and techniques.
   - Guided Instruction (Cognitivism): Step-by-step guidance from the tutor on problem-solving using RL software and tools.
   - Practical Exercises (Cognitivism/Constructionism): Hands-on implementation and experimentation with RL algorithms using software libraries and simulation environments.
   - Projects (Constructionism): Assignments and projects that require students to apply RL techniques to real-world robotic problems.

2.3 | Progression of student competencies in RL

The teaching approach in this course is designed to guide students through a progression of competencies in RL, incorporating both cognitivism and constructionism principles. Starting with simple examples and gradually moving towards more complex robotic problems, students develop a range of skills and knowledge related to RL techniques. The following section outlines the progression of competencies that students can expect to gain as they work through the examples and training methods in the course. Table 1 summarizes teaching goals and teaching approaches for each example.

2.3.1 | Understanding RL concepts and classical Q-Learning (Cognitivism)

The course begins with simple RL example, such as programming the Frozen Lake problem in MATLAB. This example serves as an introduction to RL concepts and classical Q-learning methods, which align with a cognitivism teaching approach. Students learn about the Q-value function and the update process for Q-values. By working through these example, students develop a solid foundation in RL principles and algorithms, acquiring knowledge through explicit instruction and practice.

2.3.2 | Implementing RL algorithms with classical methods (Cognitivism)

To facilitate the development and comparison of RL algorithms, we employed the OpenAI Gym library [22], a versatile toolkit for RL algorithm implementation, training, and evaluation in conjunction with simulators. OpenAI Gym can also be extended for use with real robots, providing practical testing environments for students and researchers. It encompasses various environments, such as Frozen Lake and Mountain Car, and other classical RL problems, and users can create their own environments. The Stable Baselines3 library [24] is an open-source implementation of RL algorithms in Python using the PyTorch framework. It has a single interface for all algorithms and is compatible with the Gym library and the Tensorboard library [1], which is used to monitor and visualize machine learning results.

As students progress through the course, they move on to implementing RL algorithms using software libraries and simulation environments. Students learn how to write practical programs in Python and apply classical RL

TABLE 1 List of RL examples and specific teaching goals for each example, teaching approach, and whether robot was used in example.

Example	Teaching goal	Teaching approach	Robot
Frozen Lake	Teach fundamental RL concepts: states, actions, rewards, goal, Q-learning	Cognitivism	No
Mountain Car	Transition from Q-learning to Deep RL	Cognitivism	No
Robotic tasks in simulation	Introduce simulating RL environments and implementing RL algorithms in simulations for agent movement towards a target.	Cognitivism	No
Moving to a target	Transfer RL techniques from simulation to a physical robot manipulator for controlling robot movement and navigation towards a target position in the real world.	Constructionism	Yes
Pushing ball to a target	Apply RL techniques to control a physical robot manipulator in pushing a ball towards a target position, developing effective strategies for ball manipulation using RL algorithms.	Constructionism	Yes
Pushing arbitrary object to a target	Extend RL techniques to handle pushing arbitrary objects towards a target position with a physical robot, customizing RL algorithms and reward functions for different object shapes, sizes, and properties.	Constructionism	Yes

algorithms, such as Q-learning. The examples without robots, such as the Frozen Lake and Mountain Car tasks, continue to follow a cognitivism teaching approach. Students engage in hands-on exercises to implement and experiment with classical RL algorithms using Q-tables. They receive guidance and feedback from instructors to develop their programming and problem-solving skills.

2.3.3 | Exploring deep RL (Cognitivism)

Next, students are introduced to deep Q-networks (DQN) and explore the use of deep neural networks in RL. The Mountain Car example is used to showcase the application of DQN in RL, continuing with a cognitivism teaching approach. Students learn how to approximate *Q*-values using deep neural networks, which enables them to handle problems with high-dimensional state spaces. By working on the Mountain Car example with DQN, students gain experience in implementing and fine-tuning deep RL algorithms. Instructors provide guidance and support as students explore this more complex approach.

2.3.4 | Learning deep RL models for robotic examples in simulation (Cognitivism/Constructionism)

Simulation allows for safe testing and validation of RL algorithms, facilitating algorithm development and refinement before real-world application. The simulation was divided into increasingly difficult subtasks. Students designed reward functions for desired agent actions, requiring a deep understanding of the problem domain and aligning rewards with desired outcomes. They also learned the importance of adapting environment, learning parameters, reward functions, and algorithms for optimal performance. These exercises helped students develop competencies in customizing RL techniques for specific robotic scenarios.

2.3.5 | Transferring deep RL models to physical robot (Constructionism)

In the more advanced stages of the course, students have the opportunity to transfer their RL models to physical robot manipulators, shifting completely towards a constructivist teaching approach. They focus on deep RL methods that rely on deep neural networks. The examples involving robot tasks are tackled using deep RL methods. Students are encouraged to actively engage in problem-solving, experimentation, and collaboration to adapt their RL models to physical robot systems.

2.3.6 | Adapting, evaluating, and applying deep RL techniques in robotic scenarios (Constructionism)

Throughout the course, students engage in a constructionism teaching approach to adapt deep RL techniques to specific robotic scenarios. They actively participate in

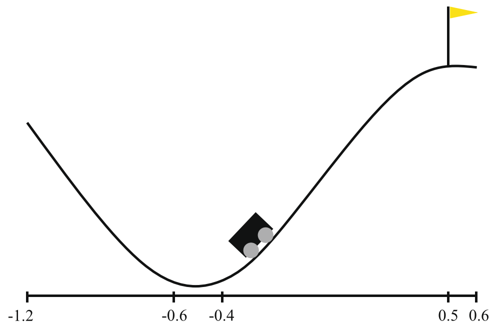

FIGURE 2 Mountain car example from the OpenAI Gym environment.

defining observations, reward functions, and actions for deep RL agents, fostering reflective thinking and creative problem-solving. Students evaluate and refine their deep RL models iteratively, aiming for improved performance and optimized learning. By the course's conclusion, students possess the skills to apply deep RL techniques effectively in diverse robotic problems, transferring models to physical robots and demonstrating competency in constructing and implementing deep RL algorithms.

2.4 | Nonrobotic teaching example: Mountain car

A car navigates a track between two peaks, aiming to reach the right mountain top (Figure 2). Due to its limited engine power, the car must build momentum by moving back and forth. The car's state is determined by its position and velocity, both continuous parameters. Three actions are available: accelerating right, accelerating left, or taking no action. Each episode ends when the car reaches the target position or after 200 actions.

To apply Q-learning and construct a Q-table, we discretize the continuous environment into 50 uniform intervals for both position and velocity. After executing the chosen action, the agent receives a reward of −1 and transitions to a new state. Negative reward of −1 acts as a penalty to incentivize the agent to reach the goal as fast as possible to achieve highest possible total reward, which in this particular example is always negative. We discretize the state before updating the Q-value using Equation (1). The episode continues until completion or until the maximum number of steps is reached. The software displays and records learning progress periodically, storing Q-tables with dimensions [50 × 50 × 3] to represent each discrete state value (position, velocity) and corresponding action.

While Q-tables are suitable for simple, discrete problems with few dimensions, more complex problems can be addressed using neural networks to approximate the *Q*-value function through deep learning. To demonstrate this, we utilize the Mountain Car example and the DQN algorithm [21] from the Stable Baselines3 library. Unlike the Q-table approach, there is no need for converting to discrete states. In Figure 3 we present graphs at different episode intervals to illustrate the agent's learning progress with classical Q-learning and learning with DQN method. Figure 3 shows the evolution of policy in various stages of learning process from very early episodes to final episodes.

2.5 | Robotic teaching examples

The robotic experiments were designed to align with the learning objectives of the course and provide students with opportunities to apply theoretical concepts in real-world scenarios. The experiments followed a progression of difficulty, starting with simple tasks and gradually increasing in complexity. The design of the experiments emphasized the integration of theoretical concepts, practical application, and problem-solving skills.

2.5.1 | Robotic system

To facilitate learning, RL algorithms are often trained in simulation environments before being transferred to real robots. Simulation enables safe testing and validation of RL algorithms [14], allowing for algorithm development and refinement before real-world application. Physics simulators provide a cost-effective and safe environment for exploring state-action spaces without physical risks [9]. The sim-to-real approach [40] bridges the gap between simulation and real-world robotics applications using RL algorithms.

The robotic teaching system consists of a collaborative Franka Emika Panda robot with seven degrees of freedom and a gripper equipped with a 3D-printed tool as an agent. A horizontal screen within the robot's workspace serves as a visualization platform for virtual and augmented reality. The screen is covered with transparent plexiglass, and an IR sensor frame surrounds the border. IR sensors detect objects placed on the plexiglass surface by measuring interruptions in the IR beams. The agent transmits its position to the robot controller. The position of the target and agent are converted in the robot coordinate system. IR sensors detect the object positions in the workspace and send the information to the agent as a state of the environment. Setup can be seen in Figures 4–6.

For a robotic task and simulation, we created an environment for object-pushing tasks. The task was

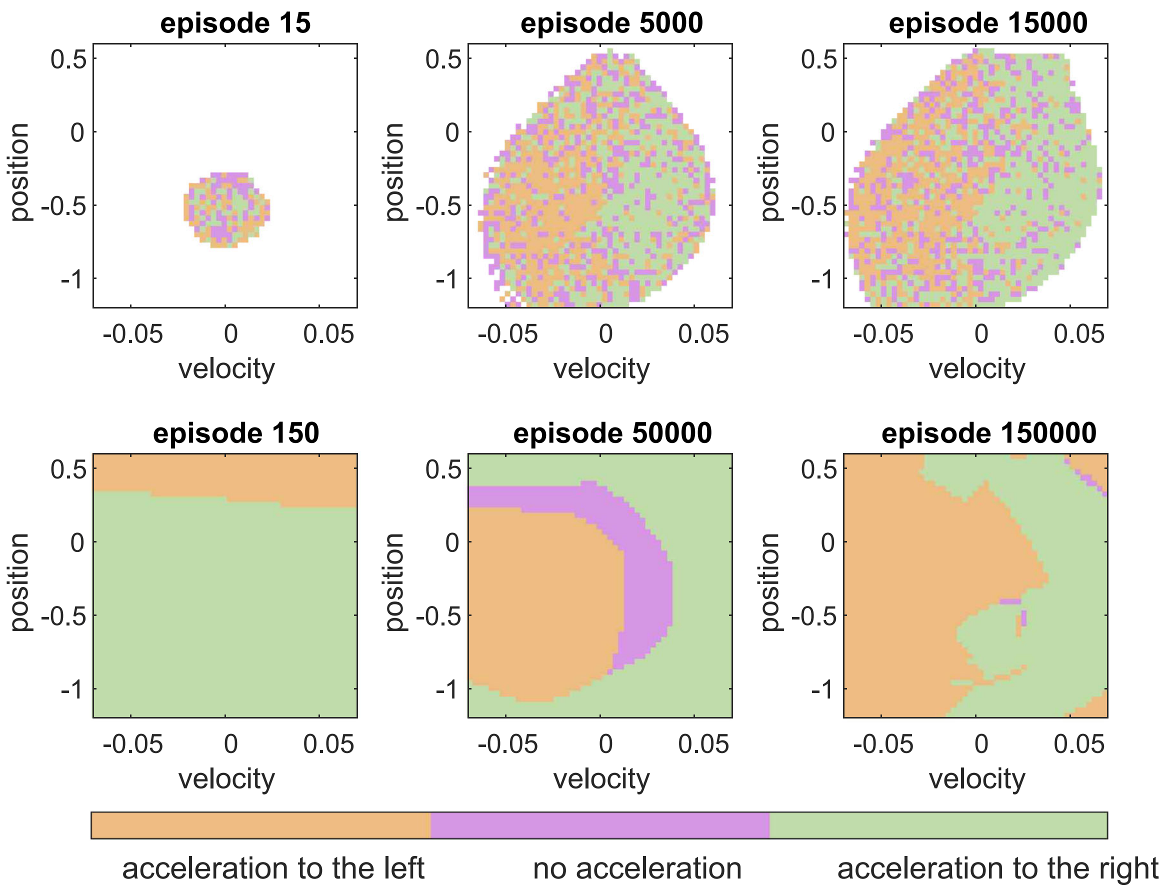

FIGURE 3 Action selection diagrams according to Q-table. First row shows diagrams for classical Q-learning for episode 15, 5000, and 15,000. White field shows unvisited states. Second row shows Q-table extracted from RL model learned with DQN method for episode 150, 50,000, and 150,000. We convert the resulting DQN model into a *Q*-value table by evaluating a discrete grid of possible states [50 × 50] and passing each state into the DQN model.

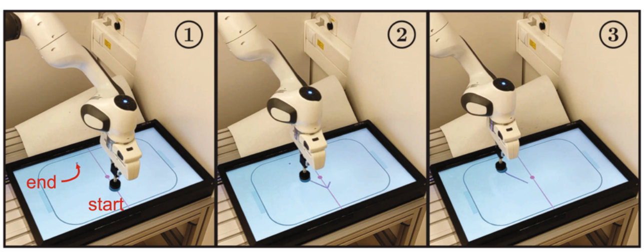

FIGURE 4 Movement of the end-effector of the robotic manipulator from the starting position to the target (small red circle) at different time instants.

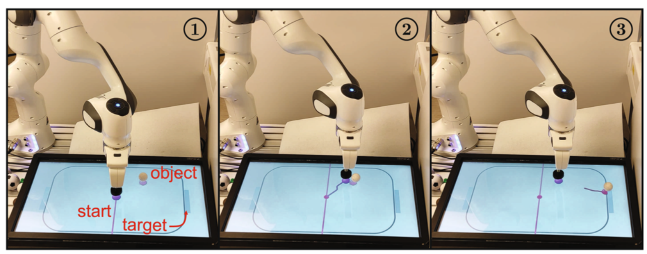

FIGURE 5 Movement of the end-effector of the robotic manipulator for the ball hitting task toward target at different instants of time.

divided into several levels with increasing difficulty. The visualization and presentation of the environment were adapted for each level. We defined the actions, while the students defined the observations and the reward function. The action set remained consistent across tasks, using a 2D velocity vector to move the agent. The observation vector dimensions varied based on the task and the number/properties of objects involved. Defining the goal as maximizing accumulated reward allowed for environment adaptation across different tasks. We implemented the tasks on the robot using the zero-shot strategy for sim-to-real transfer [40]. In this approach, the robot's end-effector's actual motion replaced the agent's simulated motion above the screen. A tool was attached to the gripper to interact with real objects.

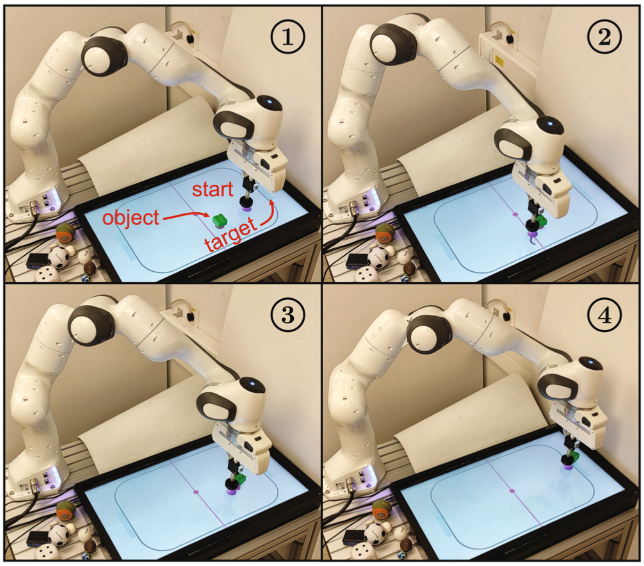

FIGURE 6 Moving the end-effector of a robotic manipulator for the task of pushing an object into the goal at different instants of time.

2.5.2 | Moving the robot to the target

The first task involves moving the robot to a randomly generated target on a screen workspace. This is achieved by sending the target and agent data to the robot controller. Figure 4 illustrates the robotic end-effector's movement toward the target.

2.5.3 | Hitting a ball into a target with the robot

The second task focuses on hitting a real ball on the screen. Based on the ball's measured position, the agent selects the appropriate action. The system was tested by randomly placing the ball on the screen before each episode. Upon program initiation, the robot manipulator moved according to the ball's position and in most cases successfully hit the ball toward the target in right edge of the screen using the tool. Figure 5 illustrates the movement of the robotic manipulator's end-effector.

2.5.4 | Pushing objects into a target with the robot

We evaluated a strategy for pushing various objects into a target position. Although this task may seem simple, it is one of the most challenging problems in robotics [20]. Traditional methods involve constructing precise friction models, understanding object geometry and characteristics, considering pushing dynamics, and incorporating these into robot control schemes [32]. This data acquisition and modeling process can be complex and time-consuming. Path-planning algorithms are often required to successfully push objects into the target. Simulation environments with physics engines can estimate physical properties and dynamics, reducing the need for detailed models. Deep learning approaches have also emerged as effective tools for learning object properties and strategies. In our experiment, we trained a robot to push objects such as balls, discs, cubes, and objects with irregular shapes into a target position. The robot successfully pushed the majority of objects to the target position without relying on friction models, object dynamics, or additional motion planning algorithms. Figure 6 illustrates the movement of the robotic manipulator's end-effector when pushing a cube into the target position. The robot maneuvered around the cube and positioned itself to push the cube toward the target, demonstrating its ability to push diverse objects without explicit models or trajectories.

2.6 | Survey

2.6.1 | Participants

The RL course was part of the Robot Control course, which spanned a full semester of 15 weeks. As detailed in this paper, the RL course was fully conducted in 2023 with 17 students participating. The RL course comprised 5 weekly sessions, each lasting 3 h. In 2022, a preliminary version of the RL course was conducted in 3 sessions without the experiments on robots described in Section 2.5, with 14 students participating in the course.

2.6.2 | Data collection

In the presented study we used qualitative and quantitative methods. The evaluation tools were:

- Anonymous questionnaire (see Table 2), given after the completion of the preliminary course in 2022 and of the full course in 2023, which documented the students' views and evaluation of their overall experience during the course. Questionnaire used a 4-grade Likert scale or Yes/No questions.
- Anonymous questionnaire (see Table 3), given before the start of the full course in 2023, which documented the students' preliminary experiences and awareness with RL. Questionnaire used a 4-grade Likert scale or Yes/No questions.

TABLE 2 Postcourse survey questions for students attending the course on RL for years 2022 and 2023.

1.	Have you heard about Reinforcement Learning before taking this course? (Yes/No)
2.	If you answered the Question 1 with Yes, how familiar were you with Reinforcement Learning before you took the course? (1—not familiar at all, 4—very familiar)
3.	To what extent has the course increased your interest and motivation to learn more about Reinforcement Learning? (1—not at all, 4—very much)
4.	To what extent do you feel confident in your understanding of the basic concepts of Reinforcement Learning? (1—not at all, 4—very)
5.	To what extent did the course convince you that you can solve complex robotic problems using Reinforcement Learning? (1—not at all, 4—very)
6.	To what extent did the course increase your confidence in solving problems with Reinforcement Learning? (1—not at all, 4—very much)
7.	How confident are you in your understanding of the impact of Reinforcement Learning in robotics? (1—not at all, 4—very much)
8.	From what you know about Reinforcement Learning, how much of that did you learn about Reinforcement Learning in this course? (1—not much, 4—all)
9.	How much did you develop your ability to create models of Reinforcement Learning during the course? (1—not at all, 4—very much)
10.	How important do you think Reinforcement Learning is in the field of robotics? (1—not at all, 4—very much)
11.	Do you agree that Reinforcement Learning can benefit robotics? (Yes/No)
12.	Did you find it challenging to experiment with Reinforcement Learning? (1—not at all, 4—very)
13.	How effective do you find this type of approach in teaching Reinforcement Learning in robotics? (1—not effective, 4—very effective)
14.	How important do you think it is to bridge the gap between theory and practical application in teaching of Reinforcement Learning? (1—not important, 4—very important)
15.	Would you recommend this course to others who are interested in learning about Reinforcement Learning in Robotics? (Yes/No)

The questions in the questionnaires were tailored to assess specific aspects related to Reinforcement Learning (RL) in robotics, such as students' familiarity with RL concepts, their motivation and interest levels, confidence in understanding RL, ability to solve complex robotic problems using RL, and perception of the impact of RL in robotics, directly linked to evaluating the effectiveness of the course on students' knowledge, skills, and perceptions regarding RL. These questions align with the research goal of investigating the impact of the course on students' understanding and application of RL techniques in robotics and can be grouped into several categories based on the aspects they explore:

- Familiarity and Understanding of RL (Q1, Q2, Q4): These questions assess students' prior knowledge and familiarity with RL concepts before taking the course, as well as their confidence in understanding the basic concepts of RL after completing the course.
- Motivation and Interest in Learning RL (Q3).
- Ability to Apply RL in Robotics (Q5, Q6, Q9): These questions focus on students' perceived ability to solve complex robotic problems using RL techniques, their confidence in solving problems with RL, and their development in creating RL models.
- Impact and Importance of RL in Robotics (Q7, Q10, Q11): These questions explore students' understanding of the impact and importance of RL in the field of robotics.
- Learning Experience and Course Effectiveness (Q8, Q12, Q13, Q14, Q15): These questions assess various aspects of the learning experience, including the amount of knowledge gained about RL, the challenge level of experimenting with RL, the effectiveness of the teaching approach, the importance of bridging theory and practice, and the likelihood of recommending the course to others.

3 | RESULTS

In 2022, 11 out of 14 students responded to the survey, while in 2023, all 17 students participated in the survey. The Mann–Whitney *U* test was used to analyze

TABLE 3 Precourse survey questions for students attending the course on RL in 2023.

1.	Have you heard of Reinforcement Learning before this course? (Yes/No)
2.	If you answered the Question 1 with Yes, how familiar are you with Reinforcement Learning? (1—not familiar at all, 4—very familiar)
3.	How interest and motivated are you to learn more about Reinforcement Learning? (1—not at all, 4—very much)
4.	If you answered the Question 1 with Yes, how confident are you that you understand the basic concepts of Reinforcement Learning? (1—not at all, 4—very)
5.	If you answered the Question 1 with Yes, can you solve complex robotic problems with Reinforcement Learning? (1—not at all, 4—very)
6.	If you answered the Question 1 with Yes, would you now be able to solve any problems with Reinforcement Learning? (1—not at all, 4—very much)
7.	If you answered the Question 1 with Yes, how confident are you in your understanding of the impact of Reinforcement Learning in robotics? (1—not at all, 4—very much)

TABLE 4 Results of the survey for questions with 4-grade Likert scale for years 2022 and 2023.

	2022		2023		
	Mean	**95% CI**	**Mean**	**95% CI**	***p*-value**
Q2	1.67	0.38	1.82	0.34	.68
Q3	3.55	0.29	3.47	0.24	.72
Q4	3.45	0.29	3.35	0.23	.62
Q5	3.00	0.44	3.41	0.29	.11
Q6	2.64	0.38	3.24	0.26	.02
Q7	3.09	0.39	3.00	0.00	.57
Q8	3.45	0.29	3.53	0.24	.72
Q9	2.45	0.46	2.94	0.26	.08
Q10	3.55	0.29	3.65	0.23	.62
Q12	2.00	0.36	2.65	0.32	.03
Q13	3.18	0.55	3.71	0.22	.13
Q14	3.73	0.36	3.88	0.15	.61

Note: Table gives mean for measure of central tendency and 95% confidence interval for measure of dispersion.

differences in responses between questionnaires for each question. A comparison of responses between the years 2022 and 2023 is presented in Table 4 and Figure 7 for questions using a 4-grade Likert scale and Table 5 compares responses to questions with Yes/No answers. In year 2022 55% and in year 2023 65% of students have heard of RL before taking the course (Q1). Students reported low familiarity with RL before the course (Q2), with average scores of 1.7 (2022) and 1.8 (2023). Students reported that the course was effective in increasing their interest and motivation to learn more about RL (Q3), with an average score of 3.6 (2022) and 3.5 (2023). Students felt confident in their understanding of basic RL concepts after the course (Q4), with average scores of 3.5 (2022) and 3.4 (2023). The course also convinced students that they could solve complex robotic problems using RL (Q5), with average scores of 3.0 (2022) and 3.4 (2023). In 2022, the course slightly to moderately increased students' confidence in problem-solving with RL (Q6) with an average score of 2.6, while in 2023, students reported a moderate increase in confidence (average score of 3.2). Students recognized the impact of RL in robotics (Q7) with average scores of 3.1 (2022) and 3.0 (2023). The majority of students felt they had learned a great deal about RL in the course (Q8), with average scores of 3.5 for both years. Students reported a slight to moderate level of development in their ability to create models of Reinforcement Learning (Q9) in year 2022 (average score of 2.5) and moderate level of development in year 2023 (average score 2.9). Students acknowledged the significant importance of RL in robotics (Q10) with average scores of 3.6 (2022) and 3.7 (2023). All students agreed that RL can benefit robotics (Q11). Experimenting with RL (Q12) was perceived as relatively easy in 2022 (average score of 2.0) but somewhat more challenging in 2023 (average score of 2.7). The course was found to be moderately effective in teaching RL in robotics (Q13) in 2022 (average score of 3.2) and moderately to very effective in 2023 (average score of 3.7). Students recognized the importance of bridging the gap between theory and practical application in RL education (Q14), with average scores of 3.7 (2022) and 3.9 (2023). Finally, when asked whether they would recommend the course to others interested in learning about Reinforcement Learning in robotics (Q15), all students responded positively, with a score of 100% for both years. This

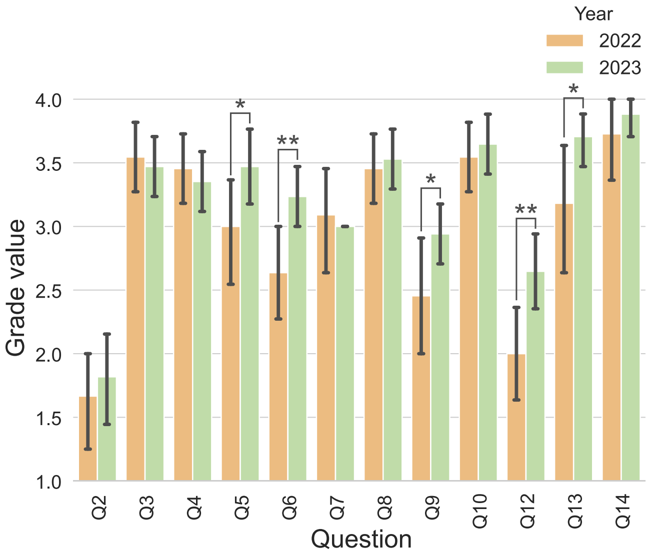

FIGURE 7 Results of the postcourse survey for questions with 4-grade Likert scale for years 2022 and 2023. Bar height represents mean and whiskers represent 95% confidence interval for measure of dispersion. Symbol * represents pair with p-value below 0.15 and symbol ** represents pair with $p < 0.05$.

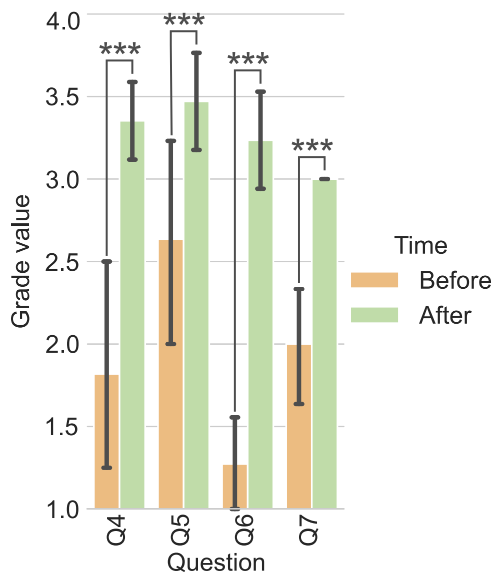

FIGURE 8 Results of the survey for questions with 4-grade Likert scale for years 2023 before the beginning of the course and after the finish of the course. Bar height represents mean and whiskers represent 95% confidence interval for measure of dispersion. Symbol *** represents pair with $p < 0.01$.

TABLE 5 Results of the survey for yes/no questions for years 2022 and 2023.

	2022	2023
Q1	55%	65%
Q11	100%	100%
Q15	100%	100%

Note: Table gives percentage of responses with yes answer.

TABLE 6 Results of the survey for questions with 4-grade Likert scale for years 2023 before the beginning of the course and after the finish of the course.

	Before		After		
	Mean	**95% CI**	**Mean**	**95% CI**	***p*-value**
Q3	3.47	0.29			
Q4	1.82	0.61	3.35	0.23	6.3E−04
Q5	2.64	0.63	3.47	0.29	3.5E−02
Q6	1.27	0.26	3.24	0.26	6.3E−06
Q7	2.00	0.36	3.00	0.00	1.3E−05

Note: Table gives mean for measure of central tendency and 95% confidence interval for measure of dispersion.

outcome indicates that students highly recommend the course to others who wish to delve into the realm of Reinforcement Learning in the context of robotics. The responses to Q6 and Q12 were statistically different with a p-value threshold of 0.05, while responses to Q5, Q9, and Q13 showed noticeable but not statistically significant differences with p-values below 0.15 (Figure 8) (Table 6).

To evaluate the course's impact in year 2023, anonymous questionnaires were used to compare students' responses before and after the course. Before the course, students expressed high interest and motivation to learn more about RL (Q3: average response of 3.5). After the course, significant improvements were observed in students' confidence in understanding RL concepts (Q4: average response increased from 1.8 to 3.4) and their self-perceived ability to solve robotic problems with RL (Q5: average response increased from 2.6 to 3.5). Additionally, students reported a substantial increase in confidence in solving problems with RL (Q6: average response increased from 1.3 to 3.2) and a better understanding of the impact of RL in robotics (Q7: average response increased from 2.0 to 3.0). These findings indicate the positive influence of the course on students' knowledge and perception of Reinforcement Learning. The differences in responses to Q4–Q7 were statistically significant with p-values below 0.01.

4 | DISCUSSION

The first objective was to investigate the effectiveness of the course students' understanding of RL concepts and their ability to apply RL techniques to solve robotic problems.

The research methodology involved conducting pre-course and post-course questionnaire to measure students' self-reported understanding and application of RL concepts. The results obtained from these assessments were compared to identify any significant differences or improvements after completing the course.

Before the course, only half to two-thirds of the students had heard of RL (Q1), and their familiarity with RL concepts was generally low (Q2). This shows that as Reinforcement Learning (RL) continues to gain traction in multiple industries, the importance of educational initiatives designed to deepen students' understanding of this field becomes increasingly evident.

The findings revealed a statistically significant improvement in students' understanding of RL concepts and their ability to apply RL techniques (Q4–Q7). The mean scores in the post-course questionnaire were significantly higher than those in the pre-course questionnaire, indicating a positive impact of the comprehensive course on students' learning outcomes. These findings align with previous studies that have evaluated the impact of RL courses on student learning outcomes. For example, Tenor et al. [19] found similar improvements in students' understanding and application of RL concepts after completing a similar course. This consistency suggests the robustness of the observed effects and the value of incorporating comprehensive RL courses in robotics curricula. Several factors may have contributed to the observed outcomes. The teaching methods used in the course, such as hands-on exercises, simulations, and real-world case studies, likely facilitated students' engagement and comprehension of RL concepts. The comprehensive nature of the course content, covering both theoretical foundations and practical applications, gradual increase of difficulty, provided students with a fundamental understanding of RL in robotics.

Post-course, students felt that they had learned a great deal about RL (Q8), demonstrating the effectiveness of the course in conveying fundamental knowledge about RL. There was a strong consensus among students in both 2022 and 2023 that RL can significantly benefit robotics (Q11), and the average rating of the perceived importance of RL in robotics was high (Q10). The importance of bridging the gap between theory and practical application was also widely recognized by students (Q14), indicating that the integration of theory and practice is an essential aspect of effective RL education. In terms of an overall course effectiveness, students perceived the course to be moderately to very effective in teaching RL in robotics (Q13). The high recommendation rate (Q15) further confirms that the students valued the course highly and found it to be effective and beneficial in learning RL for robotics.

The second objective was to investigate the impact of including a physical robot manipulator in the Reinforcement Learning (RL) education of students, compared to using simulations alone. The results show that the average scores for Q6, Q9, and Q12 in 2022 were relatively low. RL is a complex and challenging topic, and the responses from the students indicated that it might take more time and practical work to fully develop skills to use RL. While the short course in 2022 may have provided a solid foundation for understanding the basic concepts of RL, it may not have been enough to develop students' confidence in solving problems with RL or their ability to create models. Results indicated that increasing the number of hours and adding a robot for testing agents in real situations would provide more opportunities for practical exercises and projects, offering sufficient opportunities for hands-on practice and experimentation with RL. This would enable students to apply RL to real-world problems and enhance their understanding of the subject. RL requires experimentation, and students may benefit from more opportunities to apply the concepts they learn in the classroom to real-world problems.

Recognizing these limitations, the course in 2023 addressed these concerns by incorporating additional hours and introducing experimentation on a robot. The extended duration allowed for more in-depth exploration of RL concepts and increased the time available for students to practice and experiment with RL algorithms. By introducing a robot for testing agents, students were able to gain practical experience in applying RL techniques to control real-world robotic systems. The design of the robotic experiments played a crucial role in enhancing students' learning outcomes. By carefully structuring the experiments to align with the learning objectives and gradually increasing the complexity of the tasks, students were able to develop a deeper understanding of RL concepts and their practical applications. This hands-on experience likely contributed to the significant improvement on average scores for Q6 and Q12, and noticeable increase of score in Q9 in 2023. The differences between the years 2022 and 2023 highlight the importance of incorporating practical exercises with a robot. These adjustments not only addressed the shortcomings identified in the 2022 course but also improved students' learning outcomes, allowing them to better apply RL techniques to real-world problems in the field of robotics.

In educational research, especially when the number of participants is determined by class size, sample sizes can be relatively small. This limitation must be acknowledged, but the insights gained are still valuable for understanding educational interventions. To ensure the robustness of our findings, we calculated Cohen's d to analyze effect size. For questions Q6 and Q12 ($p < .05$), Cohen's d was 1.02 and 0.99, respectively, indicating a large effect size (Cohen's $d > 0.8$). For the near-significant p-values for Q5, Q9, and Q13 ($p < .15$), Cohen's d ranged from 0.71 to 0.77, indicating a medium effect size (Cohen's d between 0.5 and 0.8). These findings suggest that while the sample size is sufficient for detecting large effect sizes in Q6 and Q12, a larger sample size would be needed to confirm significant differences for Q5, Q9, and Q13. Nevertheless, the observed trends are worth noting.

The effectiveness of an RL course in robotics depends on several factors that turn theoretical knowledge into practical application. A well-designed curriculum engages students in coding RL algorithms, training robotic agents, and analyzing performance, thereby connecting theoretical concepts to real-world applications. Integrating cognitivism and constructionism, the curriculum provides an engaging learning experience, following a step-by-step approach that increases in difficulty and combines guided instruction with individual problem-solving to enhance students' understanding and application of RL techniques [7, 8, 18, 23, 28]. Clear and accessible explanations of RL concepts help students build a solid understanding, aligning with cognitivism principles [23]. Initially, guided instruction from the tutor enables students to acquire technical skills and familiarity with RL algorithms and tools. Hands-on practice and experimentation further reinforce students' understanding, starting with simple environments and advancing to more complex tasks [2]. Incorporating constructionism, the curriculum encourages students to actively engage in constructing meaningful applications in robotics [3, 12, 18, 19]. The second part of the curriculum emphasizes independent problem-solving with the tutor providing guidance rather than explicit instructions. This approach encourages active engagement, enhances problem-solving skills, and promotes learning from mistakes, offering a structured learning path that transitions from a strong theoretical and practical foundation to applying knowledge in novel and uncertain situations.

Another influential factor is the availability of appropriate learning resources and tools. Providing students with access to relevant software libraries, simulation environments, and hardware platforms enhances their learning experience. Popular RL libraries like OpenAI Gym and Stable Baselines 3, with their extensive documentation, tutorials, and code examples, can be utilized in the course.

The role of the instructor and teaching staff is also significant. A knowledgeable and experienced instructor who effectively communicates complex concepts and guides students in RL implementation is crucial. Clear explanations, practical demonstrations, and timely feedback from the instructor enhance the learning process.

While our course design yielded positive results, several limitations were identified. First, while basic methods such as Q-learning using Q-tables allow students to gain a solid understanding of the core concepts of RL, more advanced methods that utilize deep learning obscure the internal workings. Consequently, students primarily learn how to apply these advanced methods without developing a deep understanding of their inner mechanics. Second, the sim-to-real gap remains a significant challenge. Students can conduct extensive experimentation in simulation environments; however, replicating the same amount of experimentation on a real robot is not feasible due to practical constraints. For instance, the positions of the agent in simulation were directly used to move the real robot, which does not always translate seamlessly due to differences in physical dynamics and environmental factors. Third, due to the high costs and complexity of anthropomorphic robots, students had to take turns working and testing on the robot. To simplify the sim-to-real gap and provide more hands-on opportunities, less expensive and less complex robots with 2D configurations could be employed. Furthermore, limited access to high-performance computing resources can constrain students' ability to experiment with computationally intensive RL methods. Therefore, as discussed in the Introduction, the use of research and industrial robots presents technical challenges, including high costs, limited access, and operational complexity. These factors limit the extent to which students can engage with the physical hardware, potentially affecting their practical learning experience. Increasing the availability and accessibility of real-world robotic experimentation opportunities would enhance the overall learning experience.

The course's effectiveness is determined by curriculum design, resource availability, instructor expertise, and the overall learning environment. By equipping students with a fundamental understanding of RL concepts and the ability to apply them to robotic problems, such courses can significantly prepare students for real-world challenges in robotics.

5 | CONCEPTUAL GUIDELINES FOR TEACHING RL

The use of real robots in teaching RL through a cognitivism-constructivist STEM approach [3, 6, 12, 19] is a highly promising method. The customizable RL models enable a wide range of experimentation opportunities for students. This environment is also customizable to fit the specific needs in terms of equipment and knowledge level. We recommend implementing certain conceptual guidelines to aid in developing various study materials and experiments.

1. Clearly explain the concept of RL, its applications, and the relevant algorithms such as Q-learning, DQN, and policy gradient methods.
2. Provide students with hands-on experience through simulations in a controlled environment and allow them to experiment with different environments and observe the effects of changing parameters on the learning process.
3. Provide real-world examples of RL applications, such as through a robot, and encourage students to work on a short project related to RL on a real robot.
4. Provide ongoing feedback and guidance to students throughout the learning process to help them understand the material and troubleshoot any issues.
5. Promote an interactive learning environment by encouraging questions, discussions, and opportunities for students to present and share their work with the class.

6 | CONCLUSIONS

AI and RL are vital topics in robotics education. RL is a fundamental method for training intelligent robots, and students require a solid understanding of this topic. Hands-on experience through simulations and real-world examples can enhance the learning experience, making these concepts more understandable. Providing opportunities for students to work on projects related to RL on real robots can further solidify their understanding. Our course specifically focuses on RL due to its critical role in AI and robot control. In response to the fast-paced advancements in AI and deep learning, we plan to expand our curriculum to include a broader range of AI topics, which will cover advanced topics such as reasoning, perception, and environment understanding. The long-term plan ensures that our students receive a comprehensive education that prepares them for the diverse challenges in the field of robotics.

Teaching approach presented in the paper combines theoretical concepts, simulations, and real-world applications using a physical robot in an RL course. The literature survey highlights the often overlooked inclusion of robots in curricula, reinforcing the importance of integrating practical robot work into educational settings. Through a detailed description of the process of integrating simulations and real-world experiments using a physical robot, the paper provides a concrete example of addressing this gap. The research provides empirical evidence supporting the effectiveness of the proposed teaching approach, demonstrating enhancements in students' understanding and application of RL concepts. Additionally, by presenting a case study of using a physical robot in an RL course, the paper contributes to filling this gap in the literature and offers insights for educators interested in similar implementations.

A survey showed that the RL course was effective in increasing students' interest and motivation in learning about RL and its application in robotics. The positive feedback and recommendation of the course emphasize the importance of teaching RL in robotics and its potential impact.

ACKNOWLEDGMENTS

The authors acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0228).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Janez Podobnik https://orcid.org/0000-0001-9359-0297

REFERENCES

1. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, arXiv (2015).
2. J. Adams and S. Turner, *Problem solving and creativity for undergraduate engineers: Process or product?* EE 2008 - International Conference on Innovation, Good Practice and Research in Engineering Education, 2008.
3. D. Alimisis, *Designing robotics-enhanced constructivist training for science and technology teachers: the terecop project*, Proceedings of EdMedia + Innovate Learning 2008 (J. Luca and E. R. Weippl, eds.), Association for the Advancement of Computing in Education (AACE), Vienna, Austria, 2008, pp. 288–293.
4. F. Alnajjar, C. Bartneck, P. Baxter, T. Belpaeme, M. Cappuccio, C. Di Dio, F. Eyssel, J. Handke, O. Mubin, M. Obaid, and N. Reich-Stiebert, *Robots in education: An introduction to high-tech social agents, intelligent tutors, and curricular tools*, Routledge, 2021.
5. K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, *Deep reinforcement learning: a brief survey*, IEEE Signal Proc. Mag. **34** (2017), no. 6, 26–38.
6. M. Barak and M. Assal, *Robotics and stem learning: students' achievements in assignments according to the p3 task taxonomy—practice, problem solving, and projects*, Int. J. Technol. Des. Educ. **28** (2018), 121–144.

7. M. U. Bers, L. Flannery, E. R. Kazakoff, and A. Sullivan, *Computational thinking and tinkering: exploration of an early childhood robotics curriculum*, Comput. Educ. **72** (2014), 145–157.
8. C. Chalmers, M. Carter, T. Cooper, and R. Nason, *Implementing "big ideas" to advance the teaching and learning of science, technology, engineering, and mathematics (stem)*, Int. J. Sci. Math. Educ **15** (2017), 25–43.
9. J. Collins, S. Chand, A. Vanderkop, and D. Howard, *A review of physics simulators for robotic applications*, IEEE Access. **9** (2021), 51416–51431.
10. O. Coşkunserçe, *Comparing the use of block-based and robot programming in introductory programming education: effects on perceptions of programming self-efficacy*, Comput. Appl. Eng. Educ. **31** (2023), no. 5, 1234–1255.
11. B. Curto and V. Moreno, *Robotics in education*, J. Intell. Robot. Syst. **81** (2015), 3–4.
12. A. Dobrosovestnova, Constructivism in educational robotics. interpretations and challenges, Ph.D. thesis, Universität Wien, 2019.
13. N. N. Jamal, D. N. A. Jawawi, R. Hassan, R. Mohamad, S. A. Halim, N. A. Saadon, M. A. Isa, and H. N. A. Hamed, *Adaptive learning framework for learning computational thinking using educational robotics*, Comput. Appl. Eng. Educ. (2024), e22732.
14. C. F. Joventino, R. d.A. A. e. Silva, J. H. Pereira, J. M. S. C. Yabarrena, and A. S. de Oliveira, *A sim-to-real practical approach to teach robotics into K-12: a case study of simulators, educational and DIY robotics in competition-based learning*, J. Intell. Robot. Syst. **107** (2023), no. 1, 14.
15. M. A.-M. Khan, M. R. J. Khan, A. Tooshil, N. Sikder, M. A. P. Mahmud, A. Z. Kouzani, and A.-A. Nahid, *A systematic review on reinforcement learning-based robotics within the last decade*, IEEE Access. **8** (2020), 176598–176623.
16. J. Kober, J. A. Bagnell, and J. Peters, *Reinforcement learning in robotics: a survey*, Int. J. Robot. Res. **32** (2013), no. 11, 1238–1274.
17. J. Leoste, L. Jõgi, T. Õun, L. Pastor, J. San Martín López, and I. Grauberg, *Perceptions about the future of integrating emerging technologies into higher education—the case of robotics with artificial intelligence*, Computers. **10** (2021), no. 9, 110.
18. V. Lin and N.-S. Chen, *Interdisciplinary training on instructional design using robots and iot objects: a case study on undergraduates from different disciplines*, Comput. Appl. Eng. Educ. **31** (2023), no. 3, 583–601.
19. Á. Martínez-Tenor, A. Cruz-Martín, and J.-A. Fernández-Madrigal, *Teaching machine learning in robotics interactively: the case of reinforcement learning with lego® mindstorms*, Interact. Learn. Environ. **27** (2019), no. 3, 293–306.
20. G. Martius, Real robot challenge [online] (2022). https://real-robot-challenge.com
21. V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, *Human-level control through deep reinforcement learning*, Nature. **518** (2015), no. 7540, 529–533.
22. OpenAI, Gym documentation [online] (2023). https://www.gymlibrary.dev, accessed 30 January 2023.
23. J. E. Ormrod, E. M. Anderman, and L. H. Anderman, *Educational psychology: developing learners*, Pearson, Hoboken, NJ, 2016.
24. A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, *Stable-baselines3: reliable reinforcement learning implementations*, J. Mach. Learn. Res. **22** (2021), no. 268, 1–8.
25. M. Ruzzenente, M. Koo, K. Nielsen, L. Grespan, and P. Fiorini, *A review of robotics kits for tertiary education*, Proc. Int. Worksh. Teach. Robot. Teach. Robot. Integr. Robot. School Curriculum, 2012, pp. 153–162.
26. A. L. M. Santana and R. de Deus Lopes, *Active learning methodologies and industry 4.0 skills development - a systematic review of the literature*, 2020 XV Conferencia Latinoamericana de Tecnologias de Aprendizaje (LACLO) **2**, 2020, pp. 1–10.
27. I. H. Sarker, *Machine learning: algorithms, real-world applications and research directions*, SN Comput. Sci. **2** (2021), no. 3, 160.
28. D. H. Schunk, Learning theories an educational perspective, Pearson Education, Inc., 2012.
29. L. Silva Marques, C. Gresse von Wangenheim, and J. Hauck, *Teaching machine learning in school: a systematic mapping of the state of the art*, Inform. Educ. **19** (2020), 283–321.
30. K. Sivamayil, E. Rajasekar, B. Aljafari, S. Nikolovski, S. Vairavasundaram, and I. Vairavasundaram, *A systematic study on reinforcement learning based applications*, Energies. **16** (2023), no. 3, 1512.
31. N. Spolaôr and F. B. Benitti, *Robotics applications grounded in learning theories on tertiary education: a systematic review*, Comput. Educ. **112** (2017), 97–107.
32. J. Stüber, C. Zito, and R. Stolkin, *Let's push things forward: a survey on robot pushing*, Front. Robot. AI. **7** (2020).
33. R. Suenaga and K. Morioka, *Development of a web-based education system for deep reinforcement learning-based autonomous mobile robot navigation in real world*, 2020 IEEE/SICE Int. Symp. Syst. Integr. (SII), 2020, pp. 1040–1045.
34. R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*, MIT Press, Cambridge, MA, 2018.
35. T. Toivonen, I. Jormanainen, and M. Tukiainen, *An open robotics environment motivates students to learn the key concepts of artificial neural networks and reinforcement learning*, Robotics in Education, 2017, pp. 317–328.
36. E. Tosello, N. Castaman, S. Michieletto, and E. Menegatti, *Teaching robot programming for industry 4.0*, Educational Robotics in the Context of the Maker Movement (M. Moro, D. Alimisis, and L. Iocchi, eds.), Springer International Publishing, Cham, 2020, pp. 107–119.
37. S. Tselegkaridis, and T. Sapounidis, *Exploring the features of educational robotics and stem research in primary education: a systematic literature review*, Educ. Sci. **12** (2022) no. 5, 305.
38. O. Zawacki-Richter, V. Marín, M. Bond, and F. Gouverneur, *Systematic review of research on artificial intelligence applications in higher education -where are the educators?* Int. J. Educ. Technol. High. Educ. **16** (2019), 1–27.
39. T. Zhang and H. Mo, *Reinforcement learning for robot research: a comprehensive review and open issues*, Int. J. Adv. Robotic Syst. **18** (2021), no. 3.
40. W. Zhao, J. P. Queralta, and T. Westerlund, *Sim-to-real transfer in deep reinforcement learning for robotics: a survey*, 2020 IEEE Symp. Series Comput. Intell. (SSCI), 2020, pp. 737–744.

41. B. Zhong, S. Kang, and Z. Zhan, *Investigating the effect of reverse engineering pedagogy in k-12 robotics education*, Comput. Appl. Eng. Educ. **29** (2021), no. 5, 1097–1111.
42. B. Zhong, and L. Xia, *A systematic review on exploring the potential of educational robotics in mathematics education*, Int. J. Sci. Math. Educ **18** (2018), 79–101.

AUTHOR BIOGRAPHIES

Janez Podobnik is an assistant professor who received his Bachelor of Science degree in Electrical Engineering from the University of Ljubljana in 2004 and went on to obtain his PhD in the same field from the same institution in 2009. Dr. Podobnik's research interests include various applications of haptic interface, human–robot interaction, virtual and augmented reality, sensory fusion for biomechanics, and Reinforcement Learning for robotic applications. He is currently an Assistant Professor in the Laboratory of Biomedical Engineering and Robotics at the Faculty of Electrical Engineering, University of Ljubljana. In recognition of his excellent research achievements, he was awarded the Excellent in Science award for outstanding scientific achievements according to the Slovenian Research Agency in 2018.

Ana Udir received her Bachelor's degree in Electrical and Electronics Engineering from the University of Ljubljana, Faculty of Electrical Engineering in June 2019. Later, she pursued her Master's degree in Robotics from the same institution and successfully completed it in October 2022. Ana is currently working as a Product Manager at S&T Iskratel. Her role involves managing product development and strategy and collaborating with cross-functional teams.

Marko Munih received his BSc, MSc, and DSc degrees in electrical engineering from the University of Ljubljana, Slovenia, in 1986, 1989, and 1993, respectively. Initially, his research interests were focused on functional electrical stimulation of paraplegic lower extremities, including control, measurements, biomechanics, and electrical circuits. His current research focus is robot contact with the environment, haptic interfaces in the fields of industry and rehabilitation engineering, and physiological computing linked to rehabilitation robotics. Additionally, he has explored specific robot applications in construction, robots for deburring and measurement tasks, and laser technology for distance and deviation measurement. Currently, Prof. Munih is a Full Professor and Head of Laboratory of Robotics, leading the research group for Analysis and synthesis of human and machine motion. He has received numerous awards, including the Zois Certificate of Recognition in 2002 and the Vidmar Award for educational work in 2011. In 2015, he was awarded the Gold Plaquette for exceptional contributions from the University of Ljubljana.

Matjaž Mihelj graduated from the Faculty of Electrical Engineering, University of Ljubljana, Slovenia in 1996 and is currently a full professor at the Laboratory of Robotics, Faculty of Electrical Engineering, University of Ljubljana. His research interests include automation, human-robot interaction, collaborative robotics, medical robotics, robot control, and virtual and augmented reality. He was a visiting researcher at Tohoku University, Sendai, Japan in 2001, where he developed a functional electrical stimulation system for lower extremities. In 2005 and 2006, he co-developed the exoskeleton-based rehabilitation robot ARMin as a research fellow at ETH Zurich. He is the inventor of the Bimeo rehabilitation device and co-founder of the Kinestica company that develops high-tech rehabilitation devices. He has received several awards, including the Swiss Technology Award for the development of ARMin in 2006, the University of Ljubljana Rector's Award for Bimeo in 2012, and the Excellent in Science Award for outstanding scientific achievements according to the Slovenian Research Agency in 2018.

How to cite this article: J. Podobnik, A. Udir, M. Munih, and M. Mihelj, *Teaching approach for deep reinforcement learning of robotic strategies*, Comput. Appl. Eng. Educ. **32**, (2024), e22780. https://doi.org/10.1002/cae.22780