IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 July 2024, accepted 25 August 2024, date of publication 29 August 2024, date of current version 10 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3451663

RESEARCH ARTICLE

Adaptive Robot Behavior Based on Human Comfort Using Reinforcement Learning

ASIER GONZALEZ-SANTOCILDES, JUAN-IGNACIO VAZQUEZ, AND ANDONI EGUILUZ
Faculty of Engineering, University of Deusto, 48007 Bilbao, Spain

Corresponding author: Asier Gonzalez-Santocildes (gonzalez.asier@deusto.es)

This work was supported in part by the Project AI-Driven Cognitive Robotic Platform for Agile Production Environments (ACROBA) through European Union's Horizon 2020 Research and Innovation Programme under Grant 101017284, and in part by the Project EdGe Technologies for Industrial Distributed AI Applications (EGIA) through the ELKARTEK Programme from the Basque Government under Grant KK-2022/00119.

ABSTRACT This study explores the potential of training robots using reinforcement learning (RL) to adapt their behavior based on human comfort levels during tasks. An experimental environment has been developed and made available to the research community, facilitating the replication of these experiments. The results demonstrate that adjusting a single comfort-related input parameter during training leads to significant variations in the robot's behavior. Detailed discussions of the reward functions and obtained results validate these behavioral adaptations, confirming that robots can dynamically respond to human needs, thereby enhancing human-robot interaction. While the study highlights the effectiveness of this approach, it also raises the question of real-time comfort measurement, suggesting various systems for future exploration. These findings contribute to the development of more intuitive and emotionally responsive robots, offering new possibilities for future research in advancing human-robot interaction.

INDEX TERMS Community environment, human-robot interaction, learning parameters, reinforcement learning, robot behavior, task adaptation, user comfort.

I. INTRODUCTION

Today's industry is advancing rapidly, increasing the demand for integrating collaborative robots, or cobots, into work environments. These robots are designed to work alongside humans, sharing workspaces and collaborating on various tasks [1]. However, the introduction of cobots brings new paradigms and challenges not faced with traditional industrial robots, making it essential to consider factors such as work environment, human safety, and comfort [2], [3].

In collaborative robotics, it is important to understand the different levels of cooperation between human workers and robots. Figure 1 illustrates these collaborative scenarios. The Cell model represents a traditional industrial setup with distinct compartments for humans and robots. The first collaborative model, Coexistence, has humans and cage-free robots working side by side but in separate areas. The Synchronized interaction model involves humans and robots sharing a workspace but not simultaneously, ensuring coordinated workflows. In the Cooperation model, both humans and robots perform tasks in the same space simultaneously but on different products or components. Finally, the Collaboration model exemplifies the highest level of integration, with humans and robots working together on the same product or component [4].

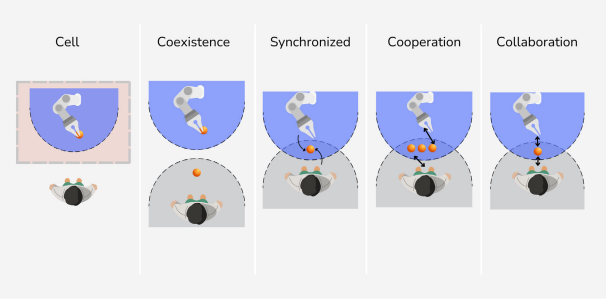


FIGURE 1. The various levels of cooperation between a human worker and a robot.

As collaborative robots become increasingly integrated into various industries, more humans are sharing their work

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Olague.

environments with robots in a cooperative manner. This shift towards shared workspaces emphasizes the importance of understanding and optimizing the interaction between human workers and cobots to ensure safety, efficiency, and overall well-being in these hybrid work settings. These emerging uncertainties have intensified research in collaborative robotics, aiming to understand the impact of humans sharing tasks with robots. Key considerations include efficiency, trust, comfort, and the emotional experiences of humans working alongside cobots [5]. Addressing these challenges is needed for developing a safe workspace [6].

A promising approach in this field is applying Reinforcement Learning (RL) to address some of these challenges [7]. RL involves an agent iteratively improving its behavior by interacting with its environment, receiving feedback through rewards and penalties to refine its decision-making process. Through RL, cobots can learn to handle simple tasks, such as lifting boxes or assembling components, and perform more complex actions [8]. However, such training requires significant time and computational resources.

The goal of applying RL in collaborative robotics is to optimize specific routines and potentially discover new behaviors. It enables direct handling of complex procedures and allows cobots to adapt to new, unforeseen situations, making real-time decisions while interacting with the environment. The ability of an RL agent to face unexpected scenarios and respond appropriately is important in collaborative robotics [9], where adaptability to evolving situations, including human reactions, is vital. These factors highlight why RL is a compelling area for shaping cobot behavior in collaborative environments [10].

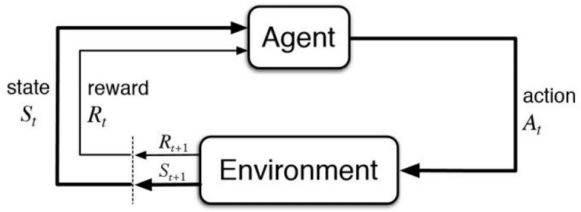


FIGURE 2. Classical RL-Loop. [11].

In classical RL, the learning process is conceptualized as a loop where an agent interacts with its environment in discrete time steps. At each step, the agent observes the current state, selects an action based on its policy, executes the action, transitions to a new state, and receives a reward signal. The agent aims to learn a policy that maximizes cumulative rewards over time. This loop of observation, action, reward, and learning, as proposed by Richard S. Sutton [11] and illustrated in Figure 2, continues until the agent achieves satisfactory performance or the environment changes significantly.

Most approaches in RL and collaborative robotics often overlook the human aspect, particularly factors such as human well-being and comfort. Integrating RL training with the human element can significantly improve the work environment and the well-being of workers in today's demanding industry. This research proposes the creation of a fully parametric environment for the scientific community, where various tasks and different robots can be trained with consideration for human comfort. This experimental setup is designed to be highly adaptable and user-friendly, allowing researchers to simulate a wide range of scenarios. Furthermore, the results obtained from these experiments can be extrapolated to real-world applications, ensuring that the findings are not only theoretical but also practically relevant.

The article is structured with an initial state-of-the-art review to set the stage and provide context. This is followed by a detailed methodology section outlining the procedures and techniques employed. Next, key aspects from the developed environmental are examined to ensure a thorough understanding before progressing to the experimentation section. After the experimentation, a comprehensive discussion of the results and future lines is presented. Finally, the article concludes with a summary of the findings.

II. STATE OF THE ART

Human-robot collaboration is a rapidly evolving field that aims to enhance safety and efficiency in various applications. Recent studies have focused on integrating reinforcement learning into human-robot collaboration to achieve this balance [12], [13]. By modifying reward structures and implementing safety-oriented shielding approaches, researchers have demonstrated significant progress towards practical solutions that can be applied in real-world scenarios [13]. These advancements in deep reinforcement learning have enabled robots to learn complex behavioral skills with minimal human intervention, leading to more autonomous and adaptive robotic systems [12].

Moreover, the development of frameworks like Multimodal Reinforcement Learning Human-Robot Collaboration (MRLC) has provided a structured approach to enhancing collaboration between humans and robots [14]. By leveraging joint-action demonstrations and encoding human user models in decision-making processes, effective teaming in collaborative tasks has been facilitated [15]. Additionally, studies have explored the use of intrinsic reward functions to enable safe interaction in industrial human-robot collaboration settings, ensuring real-time collision-free motion planning [13].

Furthermore, the concept of shared task representation and the influence of robots' fairness on human behaviors have been investigated to improve trust and cooperation within human-robot teams [16], [17]. Understanding human attitudes towards collaboration with robots is needed for shaping effective collaborative strategies and optimizing task allocation [18]. By considering factors such as human trust and decision-making processes, researchers have aimed to create fluent and efficient human-robot collaboration environments [19].

The integration of reinforcement learning, multimodal frameworks, and safety-oriented approaches in human-robot collaboration represents a significant advancement

towards achieving a harmonious balance between safety and efficiency in real-world applications.

In the context of human-robot interaction, the speed of the movements of the robot has been identified as a factor influencing human comfort. To address this, adaptation mechanisms based on reinforcement learning have been developed to interpret subconscious body signals from humans and adjust robot actions accordingly to improve overall comfort in interactions [20]. Additionally, reinforcement learning enables robots to explore and learn about their environment through positive or negative feedback, allowing them to determine favorable actions based on a reward system [21].

Reinforcement learning offers a framework for robots to learn from experience and observe the consequences of their actions on the environment, facilitating the acquisition of new skills and behaviors [22]. This learning process can be accelerated by methods such as Bilateral Biased Neighbors-Sharing Cooperative Reinforcement Learning, which integrates knowledge from neighboring robots to enhance the learning process [23].

In the realm of social robotics, reinforcement learning has been applied to adjust robot behaviors based on signals of comfort and discomfort from humans, demonstrating the potential of this approach in creating more adaptive and user-friendly robotic systems [24]. By reading subconscious body signals and using this information to adjust interaction distances, gaze meeting, and motion speed, robots can enhance their interactions with humans through reinforcement learning [25].

Overall, reinforcement learning plays a major role in enabling robots to autonomously learn, adapt, and improve their behaviors in various environments, ultimately enhancing their performance and interaction capabilities [9]. Through the utilization of reinforcement learning algorithms, robots can achieve unexpected motion patterns, exhibit good performance in tasks, and continuously improve their abilities based on feedback from the environment [26], [27]. This approach not only benefits the development of individual robots but also opens up possibilities for decentralized multi-robot systems to collaborate effectively and avoid collisions through deep reinforcement learning [28].

However, despite the promising solutions and research directions proposed in these studies, many remain theoretical and lack replicability. Developing a human-robot environment that consistently focuses on comfort variables, effectively demonstrates experimental outcomes, and is both replicable and configurable for various challenges and future research endeavors remains a significant challenge. This line of inquiry continues to be of great interest, aiming to bridge the gap between theoretical advancements and practical applications, thereby ensuring that human-robot collaboration can be reliably and consistently reproduced in real-world scenarios.

III. METHODOLOGY

This methodology section outlines the entire research process, from the creation of the environment to the experimentation and obtaining of results. Initially, a comprehensive review of scientific papers was conducted using databases such as Web of Science (WoS) and Google Scholar, among others. This review helped identify the specific needs and challenges of the research, as discussed in previous paragraphs. Based on these identified needs, the construction of a specialized experimental environment was undertaken. The following section provides a detailed description of the development of this environment, which is designed to be accessible to anyone in the scientific community.

The creation and training process of the environment has been highly iterative. Following the initial training results, minor adjustments were made to parameters, reward functions, or the environment itself to facilitate learning and correct undesirable behaviors exhibited by the robot. This cycle of training, adjustment, and result verification was repeated until the desired outcomes were achieved. Each stage of this iterative process required careful monitoring and analysis of the agents' performance, leading to incremental improvements. Future development over the next few years will follow a similar iterative process to ensure optimal results. The methodology's iterative nature ensures continuous refinement and adaptation of the system to meet evolving research needs.

This structured approach ensures a thorough exploration and refinement of the research methodology. By continuously iterating on training and adjustments, emerging challenges can be addressed, and the system's performance can be enhanced. This ongoing process of refinement is key for achieving reliable, reproducible results that can benefit the scientific community.

IV. OVERVIEW OF THE COMFORT ROBOTIC LEARNING FRAMEWORK

Everything relevant to the environment created for the experiments conducted during this investigation will be described in this section. It is reiterated that the environment is available for academic use on GitHub at the time of this article's publication [29]. The specific development framework chosen for this project is Unity. Unity was selected due to its high-quality visual simulations. Furthermore, the ability to generate simple, executable, and parametrizable environments that do not require extensive technical knowledge from external users wishing to replicate the experiments is a significant advantage. Additionally, Unity's integration with various RL libraries, such as Stable Baselines 3 [30], in addition to its native ML-Agents, makes it an excellent execution and development environment [31].

Other notable frameworks for Reinforcement Learning simulations include Gazebo and PyBullet [32], [33], known for their robust physics engines and flexibility. Gazebo, often used with ROS, is favored in research for simulating

various robotic systems [34]. PyBullet is praised for its speed and ease of use, making it suitable for quick prototyping. Despite these options being noteworthy, Unity was chosen, as previously mentioned, for its visual simulations, ease of creating executable environments, and integration with other RL libraries, making it an excellent development environment [31].

When selecting the tasks that can be performed within the environment, three types of tasks were proposed, each increasing in difficulty. The first task involves moving the robotic arm gripper to a single point, either fixed or random. The second task requires the robot to move the gripper from one point to another, following a more complex trajectory. The third task involves picking up a ball from a table and placing it on a different table based on the ball's color. These three tasks can be configured to consider the presence and comfort of a human during training. Although these three tasks are defined, for the subsequent experimentation and explanation of the environment, all examples will focus on the third task: picking up and placing a ball based on its color.

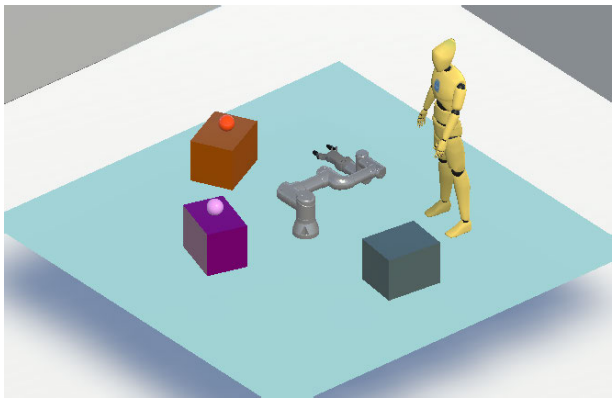


FIGURE 3. Example training environment.

In the previous Figure 3, the training environment is shown, highlighting that this is only one of the scenarios used for training. The position of the tables, the spawning of the human dummy, and the movement of the dummy are all configurable, among other parameters. There are numerous customization options available to adapt the environment to specific scenarios and training needs. Additionally, there are options to adjust parameters such as precision of the taking and leaving or division into subtasks (picking up, placing, and returning to the origin) to fit techniques like curriculum learning [35].

Regarding the division of task, the environment is designed with the capability to have a controller for the different networks and subtasks, allowing for individual training of picking up, placing, and returning. This approach aligns with the concept of hierarchical reinforcement learning (HRL), where complex tasks are decomposed into simpler, manageable subtasks [36], each with its own controller. This method can improve training efficiency and effectiveness by focusing on mastering each different task separately before integrating them. Alternatively, the entire task can be trained as a single network, without division into subtasks, to develop a comprehensive model that handles the complete sequence in one go.

Depending on the use case and the subsequent implementations that one wishes to develop or replicate, either approach can be chosen. In the research conducted, the tasks of picking up and placing based on color were carried out using a single network, while the task of returning was handled by another network. This division was chosen because, in the specific case presented, the execution time between separating the first and second tasks is practically identical. At the end of this section, all configuration parameters relevant to the experimentation will be detailed to enable replication of the presented experiments.

When it comes to comfort metrics, each user can define a different number of comfort levels depending on their measurement instruments. While this research does not focus on the measurement or selection of the number of comfort levels, it is possible to input the comfort level into the system based on external measurements collected. This flexibility allows users to tailor the comfort settings according to their specific needs and available data.

Currently, comfort in human-robot interaction considers among others, both the speed and the distance of the robot relative to the human. These two parameters have been determined by studies to directly affect how a human feels when perceiving a robot [20], [37], [38]. The reward function, which will be discussed later, takes into account not only task completion but also, if the user chooses, the speed and/or distance. The number of different comfort levels defined and the specific comfort level for each training session, as set by the user, will affect how much the robot's behavior varies. In most cases, these behaviors differ significantly between each other, as shown in the experimentation section.

Regarding the reward function, after an extensive testing period, various functions have been determined based on the task. The reward functions were carefully designed to ensure that the robot not only completes the tasks but also optimizes for efficiency and user comfort when applicable. In the equation 1, the complete reward function is represented. It includes an activation parameter k for each specific task (k_1, k_2, k_3), which enables or disables the contribution of each task's reward function based on its relevance. Additionally, the comfort function $F_{\text{comfort}}(C_l, N_c)$ is dependent on the comfort level parameter C_l and number of comfort levels N_c, ensuring that the robot's actions are adjusted according to the desired level of user comfort. The parameter N_c is a value that represents the total number of defined comfort levels in the system, while C_l is the actual comfort level the user is in. These comfort levels are designed to adjust the robot's actions according to the desired level of user comfort, ensuring a personalized and adaptive interaction. This structure allows the reward function to dynamically balance task completion,

efficiency, and user comfort.

$$R = k_1 \cdot F_{\text{task1}} + k_2 \cdot F_{\text{task2}} + k_3 \cdot F_{\text{task3}} - F_{\text{comfort}}(C_l, N_c) \tag{1}$$

The comfort function $F_{\text{comfort}}(C_l, N_c)$ is designed to ensure the robot's actions are not only efficient but also comfortable for the users. This function is based on two components: $F_{\text{conf1}}(C_l, N_c)$ and $F_{\text{conf2}}(C_l, N_c)$. The function $F_{\text{conf1}}(C_l, N_c)$ uses a hyperbolic tangent-based formula. During experimentation, linear and logarithmic functions were also tested, but the hyperbolic tangent was selected. It offers smoother transitions between penalty and reward and effectively manages threshold boundaries, leading to overall better results. The threshold for switching from penalty to reward is calculated by a function that depends on the current comfort level C_l and the total number of comfort levels N_c. To optimize for comfort, the reward is structured as $-0.1 +$ reward, ensuring that positive behaviors can cancel out the negative reward, leading to better overall results. The function considers the number of comfort levels defined by the user and the current comfort level of the user, adapting the robot's behavior to meet specific comfort requirements.

$$F_{\text{conf1}}(C_l, N_c) = -0.1 + \tanh\left(\frac{d - \theta_{\text{thr}}(C_l, N_c)}{\max(d, 1e-10)} \cdot \max(d_{\max} - d, 1e-10)\right) \tag{2}$$

where:

- d is the distance between the robot and the user.
- $\theta_{\text{thr}}(C_l, N_c)$ is the threshold for switching from penalty to reward, calculated based on the current comfort level C_l and total number of comfort levels N_c.
- $d_{\max}$ is the maximum distance for full positive reward.
- N_c represents the user-defined comfort levels and C_l the current comfort level of the user.

To better understand the mathematical function, it is helpful to provide a practical example. Suppose in a training environment, the user has defined that the robot should maintain a distance range between 20 cm and 1.2 meters (120 cm) from the user. If at a given moment, the distance between the human and the robot is 45 cm if the number of comfort levels N_c are 10 and if the user-defined comfort level C_l is 3, the result of the expression to the right of -0.1 in the formula will be negative, resulting in a total value less than -0.1. On the other hand, if the comfort level is higher, for example 6, the result of the right side of the formula will be positive, which will help to partially offset the value of -0.1 in the formula.

The idea behind this formula is to try to find a balance between the average time required to perform a task and the robot's proximity to the human, always varying depending on the comfort level. Furthermore, this approach demonstrates how, with different comfort levels, the outcome varies significantly, adjusting to the user's needs and preferences in terms of proximity and robot performance.

The function $F_{\text{conf2}}(C_l, N_c)$ is based on joints' angle, specifically considering the maximum increment per step. It defines a new increment threshold based on the comfort level, beyond which no reward is given. This adjustment ensures that the robot moves more slowly and the overall speed is reduced, improving user comfort.

$$F_{\text{conf2}}(C_l, N_c) = \begin{cases} -0.1 & \text{if } \Delta\alpha > \Delta\alpha_{\text{thr}}(C_l, N_c) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where:

- $\Delta\alpha$ is the degree increment per step.
- $\Delta\alpha_{\text{threshold}}(C_l, N_c)$ is the threshold increment per step based on the comfort level C_l and number of comfort levels N_c.

Following the logic established earlier, it is beneficial to provide a practical example to illustrate the function's operation. Assume $F_{\text{conf2}}(C_l, N_c)$ measures the joint adjustment sensitivity in a robotic system, where the robot can move its joints by up to 2 degrees every step. Consider a specific instance where the robot makes an adjustment of 0.85 degrees. If the number of comfort levels N_c are 10 and if the user-defined comfort level C_l is 4, the result of $F_{\text{conf2}}(C_l, N_c)$ will be negative, indicating an excessive adjustment beyond the user's comfort threshold. However, if the comfort level is 6, the result would be 0, signifying that an adjustment of 0.85 degrees is permissible at this higher level of comfort.

This example illustrates how $F_{\text{conf2}}(C_l, N_c)$ responds differently based on the comfort level, allowing the robotic system to adapt its behavior accordingly. The function thus serves to maximize the allowable degree of joint increment at each comfort level, ensuring that the robot's actions align closely with user preferences and comfort thresholds.

In the context of the defined mathematical model for robotic behavior, the function $F_{\text{comfort}}(C_l, N_c)$ offers a structured approach to adjust robotic actions based on user comfort levels. Following the logic outlined in previous paragraphs. Additionally, thanks to α and β, the end user can determine during the training period the weight each function should have within the overall comfort function. Through the agent's observations, the agent is capable of interacting and reacting by taking into account both parts of the comfort function.

$$F_{\text{comfort}}(C_l, N_c) = \alpha F_{\text{conf1}}(C_l, N_c) + \beta F_{\text{conf2}}(C_l, N_c) \tag{4}$$

where:

- α is a coefficient between 0 and 1 used to weight the specific importance given to $F_{\text{conf1}}(C_l, N_c)$ in the total comfort equation.
- β is a coefficient between 0 and 1 used to weight the specific importance given to $F_{\text{conf2}}(C_l, N_c)$ in the total comfort equation.

This defines the total comfort reward function, focusing primarily on the overall comfort and the results achieved. The specific functions for each task are not detailed here but

can be found on the GitHub repository. For any additional details and trial executions, everything will also be available on the GitHub repository [29]. In the final section of this paper, the future updates for the next few years will be discussed, however, one of the first planned updates following the official release is to enable variations between specific functions for each task. However, for this iteration, the emphasis remains on optimizing user comfort and analyzing the outcomes.

This section has covered the relevant aspects of the environment related to this research. The next section will present the experimental results, starting with the comfort level deactivated to ensure the proper functioning of the robotic environment in the presence of humans. Following this, the experiments will be analyzed with the comfort parameter activated to observe the variability in robot behaviors. This approach aims to bring value to the research and enhance the understanding of the environment's influence on human-robot interactions.

V. EXPERIMENTATION

In this section, the experiments conducted to validate the entire system will be discussed. Two distinct experiments were performed. The first experiment focuses on solving the environment without considering any comfort parameters. The second experiment, which will be more extensive, centers on the complete training of various executions with different levels of comfort. This includes analyzing the results, differences between them, efficiency, and other relevant factors.

Regarding the chosen algorithm, after a period of testing, SAC (Soft Actor Critic) was the most efficient algorithm [39], which is why it is normally used in continuous action trainings. To ensure the replicability of the experiments, the same hyperparameters will be used throughout all experimentation. For complete transparency and to facilitate reproduction of the results, the full configuration file is available in the repository. However, to highlight the most valuable parameters, the most relevant hyperparameters used for the SAC algorithm are listed below:

- **Learning Rate:** 0.0003
- **Batch Size:** 1028
- **Buffer Size:** 1000000
- **Tau:** 0.005
- **Hidden Units:** 256
- **Number of Layers:** 2
- **Normalize:** true
- **Gamma:** 0.99
- **Strength:** 1.0

The first of the experiments focuses on solving the environment without considering any comfort parameters. That is, according to the reward function definition from the previous sections, both α and β are set to 0. The idea of this experiment is to demonstrate that the robot in the proposed environment is capable of reacting and interacting with various situations and human movements. In this first case, no comfort parameters are taken into account. The list below provides the relevant important parameters of the third task environment necessary to replicate this specific experiment.

- **Precision:** High (10 cm)
- **Human Movement:** Random Forward Walk
- **Human Spawn Position:** Default
- **Table Position:** Default - not changing
- **Human Movement per Step:** 0.5 m
- **Simultaneous Environments in Training:** 5
- **Active Tasks in Training:** 1-2 then 3 separately
- **Clipping:** Disabled
- α **:** 0
- β **:** 0
- **Comfort Levels:** None
- **Current Level:** N/A

In the Figure 4, various situations can be observed, two where the robot has to place the orange ball and two where it has to place the pink ball. In each of these after-training testing situations, a real human has been controlling the virtual human and the movement, and it is noticeable that in the same task (placing a ball of a specific color), the robot has taken different trajectories, always trying to adapt to the human's movements. The two upper images depict the same human position, unforeseen in training, but different ball color, same for the two lower images. In these cases different trajectories can be seen. It is important to note that in more critical situations, where the human directly seeks collision, the robot is capable of adjusting and avoiding it. However, in certain critical situations, it is possible that the task may not be completed (leaving the ball).

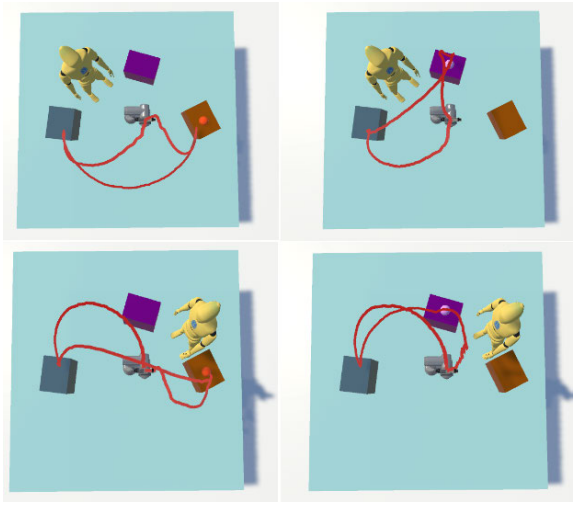


FIGURE 4. Visual representation of the results: trajectory adjustment.

This initial development and experimentation demonstrates that the robot is capable of executing a complex task. Throughout this process, the robot takes into account the human's position to avoid collisions, selects different

trajectories (marked in red in each image), and is able to modify these trajectories in real-time based on the human's movements. Considering all these aspects, the robot shows adaptability. With an even more optimal configuration of the environment and algorithm parameters, the results can be further improved. However, the main objective of this research is not only to ensure the robot's adaptability to the human's position but also to their comfort levels. This aspect is what the next set of experiments aims to test, which will be presented in the following paragraphs.

For the next experiment, it is important to provide more context. In this case, comfort will indeed be taken into account. However, for the showcase and demonstration of proper functionality, the primary function will be set with $\alpha = 1$ and $\beta = 0$. This will allow us to directly observe the correlation and analyze in detail metrics such as average distances, number of steps taken, and varying comfort levels. The goal of these results is to evaluate overall execution metrics which aims to accurately demonstrate the results.

In this case, it is also necessary to provide the training parameters to allow replication of the experiments (SAC parameters are the same as the previous experiment). The precision will be set to 30 cm to obtain results more directly, as these individual training sessions take longer to converge due to the added complexity of considering comfort. The following list details the most relevant training parameters trainings where 10 comfort levels are taken into account.

- **Precision:** Low (30 cm)
- **Human Movement:** Random Forward Walk
- **Human Spawn Position:** Default
- **Table Position:** Default - not changing
- **Human Movement per Step:** 0.5 m
- **Simultaneous Environments in Training:** 5
- **Active Tasks in Training:** 1-2
- **Clipping:** Disabled
- $\boldsymbol{\alpha}$ **:** 1
- $\boldsymbol{\beta}$ **:** 0
- **Comfort Levels:** 10
- **Current Level:** Going through 1 to 10 to verify results

In this set of experiments, the same machine as previously described was used. Each specific training session for each of the comfort levels took approximately 12 million steps, which is about 1.5 days. The time required is less than the previous experiment due to a lower precision of picking and placing the ball. The main idea is to appreciate the differences and validate the functionality of the implemented reward function. From the tests conducted, for training with a precision close to 10 centimeters, the number of steps required would be around 50 million to obtain correct results. It is recommended not to directly transition to such high precisions as 10 centimeters; instead, applying techniques like curriculum learning is the most appropriate alternative.

After completing the training sessions, the initial analysis will focus on the overall results. Specifically, from 100 executions with the trained models, the goal is to evaluate the average distance to the human during task execution (whether it involves picking up the orange or pink ball) and the average number of steps taken to complete the task. This analysis will demonstrate, in a non-visual manner, that the robot adjusts its distance based on the level of comfort and selects a route requiring a greater number of steps. The results can be seen in the following two images, one representing the average distance and the other showing the average steps in task execution throughout the environment. All executions have been conducted with the same seed, and the human's movement has been identical for each position, ensuring a fair comparison. Additionally, some test positions have never been encountered during training, thus ensuring robustness.

In the two related images (See Figures 5 and 6), a clear trend in the results of the training sessions can be observed. As the comfort level increases, meaning that the user is more comfortable with the actions of the robot, the average distance that the robot maintains with the human while performing its task decreases, meaning the task is carried out in a closer manner whenever possible. The result in this case indicates that from the first comfort level to the last, the distance decreases by approximately 25%. On the other hand, the time it takes to complete the task, measured by the number of steps, decreases as the comfort level rises, with the number of steps needed by the robot decreasing by around 12%.

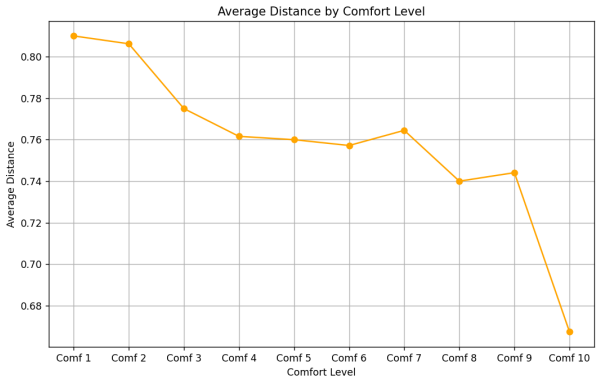


FIGURE 5. Average distance by comfort level (100 trials). Overall results.

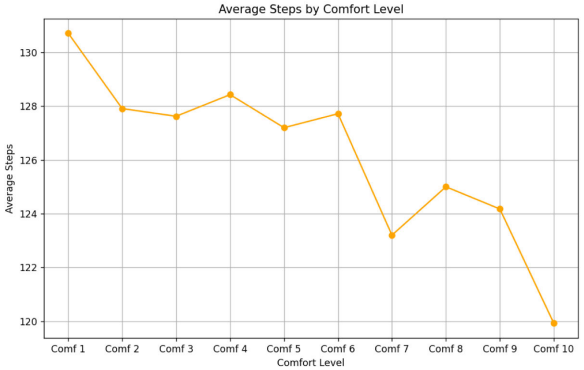


FIGURE 6. Average steps by comfort level (100 trials). Overall results.

The results are promising and demonstrate that the robot can adapt its behavior based on the comfort levels defined

by the user, adjusting according to the user's specified behavior parameters. While the differences between distinct comfort levels are clear, the similarity and proximity in certain cases may make it challenging to distinguish between contiguous levels. However, if the environment and algorithm parameters, comfort levels, and reward functions are better tuned to the specific task defined by the user, the results can be slightly improved. Additionally, for more specific tasks such as picking up only the pink ball or only the orange ball, the results are quite similar to the general results.

There are particular cases where the robot needs to carry out the task while the human is located quite far away, and due to the overall situation of the environment, the most optimal path is also the most distant. In these very specific instances, the variability in the distance is almost imperceptible. Although these cases are rare and specific, they do not affect the overall results.

An objective numerical analysis reveals the variation in the average distance between the robot and the user during training sessions. As in the previous experiment, it is beneficial to examine the robot's trajectory and the visual differences in its movement. The Figure 7 illustrates the same task executed with identical human movement, highlighting the robot's trajectory changes. Depending on the comfort level, the robot either increases its distance by taking more steps or decreases it, thereby demonstrating adaptability.

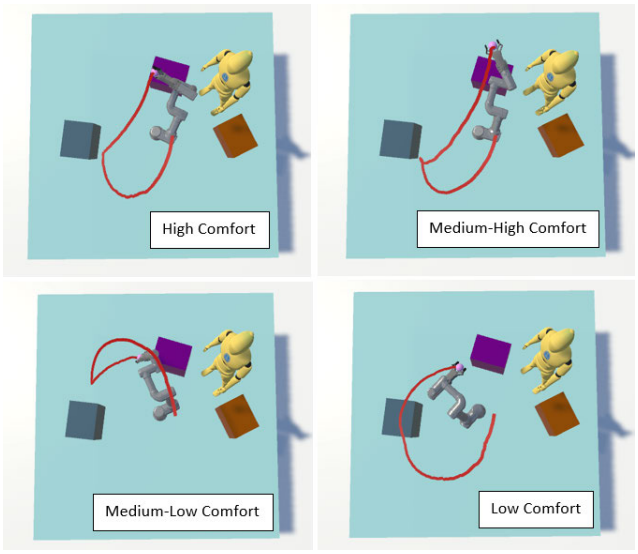


FIGURE 7. Visual Representation of the Results: comfort level trajectory adjustment.

From the previous image, it is evident that as the comfort level decreases, the robot's distance from the human increases. This can be observed across the different comfort levels: high, medium-high, medium-low, and low, corresponding to values of 10, 7, 4, and 1, respectively. These results are validated not only numerically, as shown in the graphs, but also visually.

The four images illustrate the robot's trajectory at different comfort levels. In the high comfort level image, the robot maintains a shorter distance from the human, taking a more direct path to complete the task. As the comfort level decreases, the robot's path becomes progressively longer and more curved, moving further away from the human. This visual representation clearly demonstrates how the robot adjusts its trajectory based on the specified comfort levels.

Testing various scenarios to determine if this reaction is consistent in entirely different situations is noteworthy. These include cases where the user is positioned differently, performing different movements, and where the robot is tasked with picking up a different ball. The next Figure 8 aims to visually present the results of this different task. It is evident that the robot follows distinct trajectories compared to those observed previously, and these trajectories vary according to the different comfort levels.

Figure 8 serves to validate the earlier observations and provides an additional perspective. This image not only shows the trajectory but also illustrates how the robot approaches to place the ball. In this second set of images, a substantial difference is observed between the high comfort and medium-high comfort levels.

The comparisons between high-medium, medium-low, and low show less pronounced differences, but it remains clear from the numerical data that the radius increases (The maintained distance grows slightly). Highlighting this behavior is important because, depending on the human's position, the difference in the robot's trajectory can be more pronounced, as seen in the previous set of images, or less pronounced, as observed in this set. This differentiation is needed for understanding the adaptability of the robot's behavior in varied environments and how it responds to changes in the user's location.

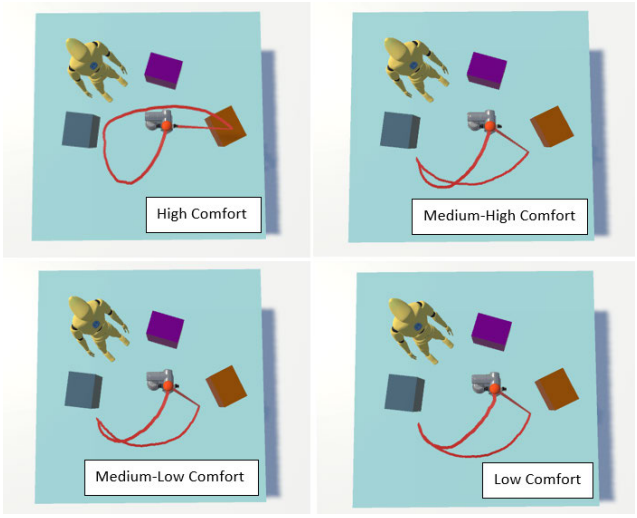


FIGURE 8. Visual representation of the results (II): comfort level trajectory adjustment.

For this exhaustive experimentation, the second function of the complete comfort function has not been considered,

with the parameter β set to 0. Additionally, based on the tests conducted to determine the correct functionality of the reward function, activating this parameter would result in a change in the number of steps, with the angular differential between steps being greater at higher comfort levels and smaller at lower comfort levels. Future versions of the framework will add new options to the comfort function, allowing more customization based on specific research needs.

In the following sections, the impact of the environment and the results will be analyzed in detail. Additionally, a vision of future implementations of the environment will be offered from the day of its publication. This will provide insights into potential improvements and extensions to enhance the framework's applicability in various research contexts.

VI. DISCUSSION

In this section, the obtained results will be discussed in detail, exploring their implications and significance. Additionally, future lines of work related to the experimentation will be outlined, along with a concrete plan for future additions to the training environment. This plan aims to provide the scientific community with a clear understanding of the anticipated advancements and improvements, ensuring transparency and fostering collaboration.

The results obtained from the human-robot collaboration environment based on comfort levels have been positive. Initially, the robot reacted to the user's position, adapting its behavior accordingly. When comfort levels were introduced, the robot exhibited distinct behaviors at each level, demonstrating its ability to adjust its actions based on the specified comfort parameters. This adaptability highlights the effectiveness of the comfort function in modulating the robot's interactions to enhance user experience.

Furthermore, extensive testing was conducted, using manual controls to simulate previously unseen situations. The robot successfully adapted to these new challenges, showcasing its robustness. Although increased training hours will undoubtedly enhance its robustness to unforeseen situations, the environment already offers various alternatives to make the learning process easier. These features, although not used in the experimental process, simplify the training, making the system more accessible for specific cases in the scientific community.

It is important to emphasize that the environment is free and available to the scientific community. Additionally, it will receive periodic updates, adapting to new technologies as they emerge. These updates will ensure the environment remains cutting-edge, versatile, and highly beneficial for the scientific community, fostering continuous innovation and practical application in various research contexts.

Future updates for the training environment include several significant advancements. In next versions, users will have the option to choose between different specific reward functions for more personalized environment settings. New robot models with their controllers will be developed, as currently only the UR3e is supported. Additionally, new environments featuring more specific and longer tasks will be introduced. There will also be a development of a graphical application for fine-tuning all parameters, which is currently done through JSON or YAML.

Final updates will focus on establishing a framework for testing the generated neural networks in virtual reality, visualized in real industrial environments with available parameterization, providing a plug-and-play solution. This development is currently in a beta phase, where it is possible to test and interact with the system, placing balls and sharing tasks with the robot. As shown in the Figure 9, this is a set of direct screenshots from the virtual reality headset, illustrating the entire environment, the interaction and the robots in action. Final adjustments are still necessary for the full release of this significant update.

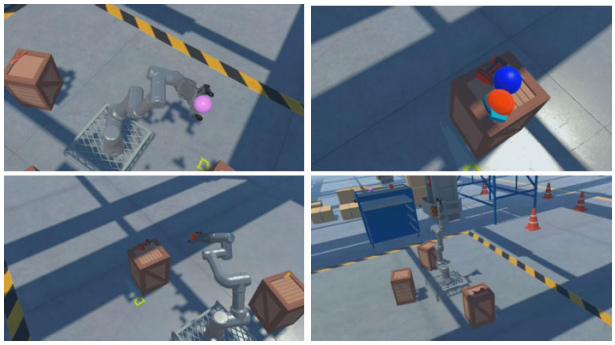


FIGURE 9. Set of 4 screenshots of the interaction between human and robot collaboration in a VR environment.

The experimentation process, combined with the training environment and its future enhancements, provides an interesting pathway for developing robots that continuously adapt to the needs and preferences of individual workers. This framework is designed to ensure that robotic systems remain responsive and personalized, enhancing both efficiency and user satisfaction. This comprehensive approach promises to deliver benefits to the scientific community and industry, promoting the development of robots that are both highly functional and aware of human needs.

VII. CONCLUSION

This conclusion section will revisit and summarize the most significant contributions of this research. Additionally, this section will propose future research directions, offering valuable insights and opportunities for other researchers to explore and build upon the presented environment. These suggested lines of investigation aim to further enhance the field and expand the practical applications of the developed framework.

The most valuable contributions of this research can be summarized by its key advancements in the field of human-robot collaboration. The study developed an adaptive framework that allows robots to adjust their behavior based on stress levels and comfort parameters, enhancing user

experience and operational efficiency. The introduction of a comprehensive comfort function, validated through rigorous testing, demonstrated the robot's ability to respond dynamically to different user-defined comfort levels. The creation of an accessible and updatable training environment, combined with future planned advancements, ensures that the framework will remain relevant and continue to evolve. These contributions collectively pave the way for more personalized and responsive robotic systems, capable of adapting to individual user needs and improving overall human-robot interaction.

Building on the proposed environment, which as mentioned, is freely available, other researchers can implement their own modifications to meet specific needs or fields if the customization options provided are not sufficient. For example, in a medical setting, the environment could be adapted to train robots to assist with patient care, optimizing comfort and stress levels for both patients and healthcare providers. In an industrial context, the framework could be customized to improve worker safety and efficiency, such as enhancing robot movements for heavy lifting tasks or increasing precision in assembly line operations. These examples show the environment's versatility, allowing it to be adapted to various applications and extending its impact across different domains.

As previously mentioned in the discussion section, creating realistic testing environments, including virtual reality simulations, is essential for advancing human-robot collaboration. The development of a virtual reality framework is currently in progress, designed to simulate real-world scenarios. This framework aims to provide a safe and controlled environment for testing robot behavior and interactions with humans. Future plans include conducting tests with actual human participants to further validate and refine the system. This approach can be highly beneficial for other researchers as well, offering a valuable line of work to develop and implement various applications, ensuring that robotic systems are robust and adaptable to real-world conditions.

A future line of work involves testing the system with real robots in actual physical environments. The framework has been designed to facilitate this transition from virtual simulation to real-world application, ensuring that the output is easily transferable to a real robot. As discussed throughout this paper, measuring comfort directly from the user is a key component. This can be achieved through various means, ranging from a simple button placed next to the user to more sophisticated methods such as heart rate sensors or EEG devices. By integrating these real-time measurements, the system can be fine-tuned to respond accurately to the user's comfort levels, thereby enhancing the effectiveness of human-robot interaction in real-world scenarios.

In summary, this research establishes a foundation for improving human-robot collaboration by developing an adaptive framework based on stress levels and comfort parameters. The experimentation demonstrates that robots are adaptable, responding effectively to different user-defined comfort levels. Future work includes adding new robot models and environments, integrating virtual reality for testing, and applying the framework in various fields such as medical and industrial settings. Although the framework is prepared for real-world deployment, successful implementation requires adapting the robot's controller with the neural network and securely preparing the real environment. These steps, combined with real-time comfort measurements, are essential to ensure effective and safe operation with physical robots.

This framework shows the potential to enhance human-robot interaction by making robots more responsive to individual user needs. Continued refinement and exploration of new research directions will contribute to developing intelligent, adaptable robotic systems that improve productivity and user satisfaction across different domains.

REFERENCES

[1] A. Weiss, A.-K. Wortmeier, and B. Kubicek, "Cobots in industry 4.0: A roadmap for future practice studies on human–robot collaboration," *IEEE Trans. Hum.-Mach. Syst.*, vol. 51, no. 4, pp. 335–345, Aug. 2021.

[2] F. Sherwani, M. M. Asad, and B. S. K. K. Ibrahim, "Collaborative robots and industrial revolution 4.0 (IR 4.0)," in *Proc. Int. Conf. Emerg. Trends Smart Technol. (ICETST)*, Mar. 2020, pp. 1–5.

[3] H. M. Parsons, "Human factors in industrial robot safety," *J. Occupational Accidents*, vol. 8, nos. 1–2, pp. 25–47, Jun. 1986.

[4] W. Bauer, M. Bender, M. Braun, P. Rally, and O. Scholtz, "Lightweight robots in manual assembly–best to start simply," Frauenhofer-Institut für Arbeitswirtschaft und Organisation IAO, Stuttgart, Germany, Tech. Rep., 2016, vol. 1.

[5] V. D. Simone, V. D. Pasquale, V. Giubileo, and S. Miranda, "Human–robot collaboration: An analysis of worker's performance," *Proc. Comput. Sci.*, vol. 200, pp. 1540–1549, Jan. 2022.

[6] D. Kragic, J. Gustafson, H. Karaoguz, P. Jensfelt, and R. Krug, "Interactive, collaborative robots: Challenges and opportunities," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 18–25.

[7] T. B. Sheridan, "Human–robot interaction: Status and challenges," *Hum. Factors*, vol. 58, pp. 525–532, Jun. 2016.

[8] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, Sep. 2013.

[9] P. Kormushev, S. Calinon, and D. Caldwell, "Reinforcement learning in robotics: Applications and real-world challenges," *Robotics*, vol. 2, no. 3, pp. 122–148, Jul. 2013.

[10] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annu. Rev. Control, Robot., Auto. Syst.*, vol. 5, no. 1, pp. 411–444, May 2022.

[11] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[12] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 3389–3396.

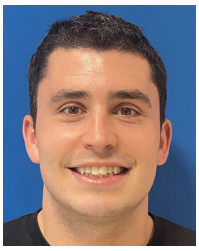
[13] Q. Liu, Z. Liu, B. Xiong, W. Xu, and Y. Liu, "Deep reinforcement learning-based safe interaction for industrial human–robot collaboration using intrinsic reward function," *Adv. Eng. Informat.*, vol. 49, Aug. 2021, Art. no. 101360.

[14] Z. Cai, Z. Feng, L. Zhou, C. Ai, H. Shao, and X. Yang, "A framework and algorithm for human–robot collaboration based on multimodal reinforcement learning," *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–13, Sep. 2022.

[15] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah, "Efficient model learning from joint-action demonstrations for human–robot collaborative tasks," in *Proc. 10th ACM/IEEE Int. Conf. Hum.-Robot Interact. (HRI)*, Mar. 2015, pp. 189–196.

[16] M. Dalmasso, J. E. Domínguez-Vidal, I. J. Torres-Rodríguez, P. Jiménez, A. Garrell, and A. Sanfeliu, "Shared task representation forlla human–robot coborative navigation: The collaborative search case," *Int. J. Social Robot.*, vol. 16, no. 1, pp. 145–171, Jan. 2024.

[17] J. Cao and N. Chen, "The influence of robots' fairness on humans' reward-punishment behaviors and trust in human–robot cooperative teams," *Hum. Factors, J. Hum. Factors Ergonom. Soc.*, vol. 66, no. 4, pp. 1103–1117, Apr. 2024.
[18] V. Villani, A. Ciaramidaro, C. Iani, S. Rubichi, and L. Sabattini, "To collaborate or not to collaborate: Understanding human–robot collaboration," in *Proc. IEEE 18th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2022, pp. 2441–2446.
[19] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Trust-aware decision making for human–robot collaboration," *ACM Trans. Hum.-Robot Interact.*, vol. 9, no. 2, pp. 1–23, Jun. 2020.
[20] W. Wang, Y. Chen, R. Li, and Y. Jia, "Learning and comfort in human–robot interaction: A review," *Appl. Sci.*, vol. 9, no. 23, p. 5152, Nov. 2019.
[21] S. W. H. Teoh, "Reinforcement learning for mobile robot's environment exploration," *J. Phys., Conf. Ser.*, vol. 2641, no. 1, 2023, Art. no. 012003.
[22] D. Koert, M. Kircher, V. Salikutluk, C. D'Eramo, and J. Peters, "Multi-channel interactive reinforcement learning for sequential tasks," *Frontiers Robot. AI*, vol. 7, p. 97, Sep. 2020.
[23] A. Iskandar and B. Kovács, "A survey on automatic design methods for swarm robotics systems," *Carpathian J. Electron. Comput. Eng.*, vol. 14, no. 2, pp. 1–5, Dec. 2021.
[24] N. Akalin and A. Loutfi, "Reinforcement learning approaches in social robotics," *Sensors*, vol. 21, no. 4, p. 1292, Feb. 2021.
[25] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, and N. Hagita, "Adapting robot behavior for human–robot interaction," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 911–916, Aug. 2008.
[26] R. Yamashina, M. Kuroda, and T. Yabuta, "Caterpillar robot locomotion based on Q-learning using objective/subjective reward," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Dec. 2011, pp. 1311–1316.
[27] M. Hara, M. Inoue, H. Motoyama, J. Huang, and T. Yabuta, "Study on motion forms of mobile robots generated by Q-learning process based on reward databases," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 6, Oct. 2006, pp. 5112–5117.
[28] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 6252–6259.
[29] A. Gonzalez. (2024). *Comfort-RL: Comfort Robotic-Learning*. Accessed: Jul. 22, 2024. [Online]. Available: https://github.com/AsierGonz/Comfort-RL
[30] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *J. Mach. Learn. Res.*, vol. 22, pp. 268:1–268:8, Jan. 2021.
[31] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange, "Unity: A general platform for intelligent agents," 2018, *arXiv:1809.02627*.
[32] Gazebo. (2014). *Open Source Robotics Foundation*. Accessed: Jun. 11, 2024. [Online]. Available: http://gazebosim.org/
[33] E. Coumans and Y. Bai. (2019). *PyBullet, a Python Module for Physics Simulation for Games, Robotics and Machine Learning*. [Online]. Available: http://pybullet.org
[34] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2020, pp. 737–744.
[35] S. Narvekar, B. Peng, M. Leonetti, J. Sinapov, M. E. Taylor, and P. Stone, "Curriculum learning for reinforcement learning domains: A framework and survey," 2020, *arXiv:2003.04960*.
[36] S. Pateria, B. Subagdja, A. Tan, and H. C. Quek, "Hierarchical reinforcement learning," *ACM Comput. Surv.*, vol. 54, pp. 1–35, Jan. 2021.
[37] M. M. E. Neggers, R. H. Cuijpers, and P. A. M. Ruijten, "Comfortable passing distances for robots," in *Social Robotics*, S. S. Ge, J.-J. Cabibihan, M. A. Salichs, E. Broadbent, H. He, A. R. Wagner, and Á. Castro-González, Eds., Cham, Switzerland: Springer, 2018, pp. 431–440.
[38] M. M. E. Neggers, R. Cuijpers, P. A. M. Ruijten, and W. Ijsselsteijn, "The effect of robot speed on comfortable passing distances," *Frontiers Robot. AI*, vol. 9, Jun. 2022, Art. no. 915972.
[39] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine, "Soft actor-critic algorithms and applications," 2018, *arXiv:1812.05905*.



ASIER GONZALEZ-SANTOCILDES was born in Bilbao, in 1999. He received the Graduate degree in computer engineering and in industrial electronics and automation engineering and the master's degree in computation and intelligent systems from the University of Deusto, Bilbao, Spain, in 2022 and 2023, respectively. He is currently pursuing the Ph.D. degree with the Technological Institute, DeustoTech.

His major field of study was artificial intelligence and emerging technologies. During his undergraduate and master's studies, he received several awards for his final degree and masters projects. He is developing his doctoral thesis in the field of artificial intelligence, robotics, and emerging technologies with DeustoTech. He is also a Lecturer with the University of Deusto, teaching courses, such as reinforcement learning, web engineering, and advanced interactive technologies. He has also worked in several projects in sectors, including renewable energy and cybersecurity.

JUAN-IGNACIO VAZQUEZ received the joint Ph.D. (Doctor Europeus) degree in computer science and artificial intelligence from the University of Deusto, Spain, and Lancaster University, U.K., in 2007. He is currently an Associate Professor with the Faculty of Engineering, University of Deusto. His research interests include reinforcement learning, multi-agent strategies, human–robot interaction (HRI), and the Internet of Things (IoT), with a particular emphasis on cognitive robotics and development of sophisticated simulation environments. He was a member of European Internet of Things Council and the Founder of several technology start-ups. From 2020 to 2023, he was the Director of the Deusto Institute of Technology—DeustoTech, where he was the Director of the Smart Objects Unit, specialized in intelligent objects connected to the internet.

ANDONI EGUILUZ received the degree in computer science engineering from the University of Deusto (UD), Spain, in 1991.

He was recognized with the best academic record of his year with UD. In 1996, he became a fellow of the Advanced Study Program with Massachusetts Institute of Technology (MIT), USA. He has been a Professor with the Computer Engineering Department, UD, since 1991, teaching subjects, such as programming, compilers, multimedia, operating systems, human–computer interaction, and accessibility. Since 2020, he has been the Head of the Computer Engineering Department, and since 2023, he has been coordinating the Response and Action to Generative Artificial Intelligence Disruption Team. His work experience includes managing gizer.net (2004–2010), a company focused on social technology and accessibility projects and directing the Computer Center, Faculty of Engineering, UD (1998–2002). His research interests include computational thinking, accessibility, and multimedia. His doctoral thesis, "Analysis of the Development of Computational Thinking With Generalization through Visual Programming Challenges," in 2020. He has an H-index of 14. He has been the Founder and the Secretary of CEPACC, since 2005, a network for research, development, and teaching on communication accessibility. He has been the President and a member of the Association for Universitarian Solidarity, ANIMO, since 2000. His current projects focus on AI in education, computational thinking, technology and audiovisual content for education, multimedia, web accessibility, and innovation.