2024

Reinforcement Learning for Optimal Kicking Actions in Humanoid Robotics: Advancing Robotic Autonomy and Versatility

Suresh Dodda
M. Tech

Sathish Kumar Chintala
M. Tech

Sukender Reddy Mallreddy
M. Tech

Sharath Chandra Macha
M. Tech

Yashwanth Vasa
M. Tech

See next page for additional authors

Original Publication Citation

Dodda, S., Chintala, S. K., Mallreddy, S. R., Macha, S. C., Vasa, Y., Bonala, S. B., Kamuni, N., & Alla, S. (2024). *Reinforcement learning for optimal kicking actions in humanoid robotics: Advancing robotic autonomy and versatility* [Paper presentation]. American Society for Engineering Management 2024 International Annual Conference, Virginia Beach, Virginia.

Proceedings of the American Society for Engineering Management 2024 International Annual Conference
G. Natarajan, H. Zhang, and E.H. Ng eds

REINFORCEMENT LEARNING FOR OPTIMAL KICKING ACTIONS IN HUMANOID ROBOTICS: ADVANCING ROBOTIC AUTONOMY AND VERSATILITY

Suresh Dodda, M.CA
Sathish Kumar Chintala, M.S.
Sukender Reddy Mallreddy, M.S.
Sharath Chandra Macha, M.S.
Yashwanth Vasa, M.S.
Sapan Bharadwaj Bonala
Navin Kamuni, M.Tech

Sujatha Alla, Ph.D.*
Old Dominion University, United States

***salla001@odu.edu**

Abstract
Acquiring the necessary skills to perform a work effectively and efficiently requires a significant investment of time and computing power. Previous applications of Reinforcement Learning (RL) for action optimization in humanoid robotics have shown how promising this technology is for moving robotics towards true autonomy and versatility. Therefore, this study offers the first use of RL to create an entirely optimal kicking action for the Alderbaran Nao robot. Kicking motions that were steady, precise, quick, and able to kick farther than any existing RoboCup squad were generated by optimizing for a multi-objective reward function. We demonstrate that the ideal kicking motions can be modified to produce angled kicks by putting a dynamic kicking module into practice. We also research on various kicking movements and more intricate search spaces that can benefit from the methodology presented in this study.

Keywords
Humanoid Robots, Optimization, Reinforcement Learning, RoboCup

Introduction
Rapid improvements in power sources and motor technology have greatly accelerated the field of robotics, allowing for the creation of smaller, more energy-efficient robots that can also carry out ever-more complex tasks. The evolution of robots has been astounding, from basic sensor-driven devices like Tortoise, which reacted to external stimuli through modifying sensor voltage, to complex robots like Honda's Asimo which demonstrates advanced skills in image recognition and environmental analysis. Asimo demonstrated its ability to carry out dynamic tasks in 2012, including running, hopping on one leg, unscrewing bottles, and pouring liquid into glasses. Each limb has a vast number of motors that support these actions, giving it many Degrees of Freedom (DoF) to enable intricate and accurate movements. With more motors and DoF, the complexity of managing these movements has increased. It used to take a lot of time and error-prone manual coding to program these movements and test the torque and voltage levels. Generally, inverse kinematics has been used in robotics to improve joint movement and provide precise control over robotic limbs and end-effectors, such as hands and feet. Still, the problem of figuring out how to move these end-effectors to complete particular jobs remains, thus we need to progress towards Artificial Intelligence (AI) (Alla et al., 2018; Alla, Soltanisehat, & Taylor 2018; Alla 2019; Lakshminarayan et al., 2023; Kamuni et al., 2024a, 2024b, 2024c) especially learning approaches, to improve and optimize these motions. Supervised learning, which depends on training data to produce exemplary solutions that may be improved and optimized, has long been the predominant method in robotics research. But there's a new approach on the horizon called Reinforcement Learning (RL) that looks promising (Kamuni et al., 2024; Kashyap et al., 2022, 2023, 2024; Kumar et al., 2023; Marwah et al., 2023; Sa al.,

2016). This strategy is modelled after the way that creatures learn naturally through rewards and punishments, which help them learn from the results of their activities. Robots that follow a similar approach are able to learn from and improve upon previous training data.

Therefore, to create a dynamic kicking engine for humanoid robots, this study examines the application of RL. Advanced robots are now more affordable (Alla & Pazos, 2019) for academic study thanks to advancements in mechatronics over the past ten years, which have also decreased the cost of robotic hardware and software. To better understand and enhance the robot's ball-kicking capabilities, this study will make use of one such robot. It will concentrate on motions that are optimized for accuracy, power, speed, stability, and stability. Our study focuses on the RL-based generation of kicking motions, with the goal of creating guidelines that specify these motions and modify them according to the angle, kicking distance, and ball position. Before adjusting to create angled kicks, the primary emphasis is on making straight kicks as effective as possible over a range of distances. The approach that was selected, Q-learning, is a kind of RL that is well-known for its effectiveness in fast convergence to optimal solutions. In spite of the restricted amount of environmental knowledge the study's hypothesis indicates that RL can successfully generate an ideal kicking motion that is suited to certain task restrictions. The long-distance ball travel, robot stability during the kick, kick execution time, and kick angle accuracy are all included into the reward function. According to tests conducted in real environments, the anticipated outcomes of this study will show that RL is a practical method for learning and refining robotic movements. The best kicks produced by this technique are expected to be very different from the normal kicks made by humans, which usually feature a strong backswing. To maximize the motor's ability to reach maximum speed, the robotic kick is anticipated to have a limited backswing instead. This study not only uses RL to tackle a realistic robotic problem, but it also demonstrates the method's wider relevance and efficacy in robotics, which could result in improved and adaptive robotic behaviors in real-world scenarios.

The study is as follows; the background is provided in the following section. The related works are presented in Section 3. The pre-implementation is provided in Section 4. The post-implementation is presented in Section 5. The experimental analysis is covered in Section 6, and Section 7 offers some conclusions and ideas for further research.

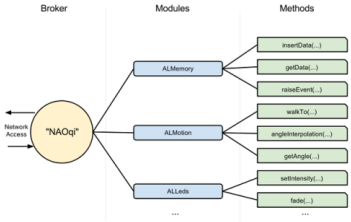
Background

To create the ideal kicking motion for the Alderbaran Nao humanoid robot—the RoboCup Standard Platform League (SPL) uses RL. Aiming to build an autonomous humanoid robot team that can overcome the human world champion team by 2050, RoboCup is an annual international robotic football competition. This program creates a realistic, competitive environment for testing and enhancing robotic skills while fostering international cooperation to progress robotics and AI. The SPL matches are played on a 6 by 4-meter pitch between two teams of four Aldebaran Nao robots, with regulations similar to human football. Goal-scoring via kicking the ball into the opponent's goal is the main objective, which is comparable to human football. But in addition to scoring goals, the robots have a kicking module that is intended to carry out strategic actions, such as passing to teammates or attempting a goal from other field locations as demonstrated in research testing but not yet effectively applied in RoboCup matches. The Nao robot, created especially for RoboCup has 21 DoF, 11 of which are in its legs, enabling it to mimic human-like kicking actions. Nevertheless, brushed Direct Current (DC) motors are used by Nao robots instead of elastic stretch and rapid release, which is how human muscles produce force. The best kicking motion that comes from RL is influenced by these motors since they provide exact control over motions but lack the elastic energy that is present in human muscle. Even so, servo motors' accuracy, which comes from its electromagnetic stepper design, is a big benefit since it offers reliable repetition in movements, which is essential for robotics.

However, complex control of limb movements is required in humanoid robotics to accomplish desired positions or actions. Forward and reverse (inverse) kinematics are the two main techniques used to control limb movements. Forward kinematics provides a simple but inflexible method by using predefined joint angles to calculate the position of an end-effector. However, because of the mechanical properties and DoF of the involved joints, inverse kinematics provides a more flexible method by determining the required joint angles to reach a particular end-effector position. This method often leads to multiple possible solutions. This adaptability is especially helpful in robotics, where exact end-effector placements are necessary. In order to guarantee that the end-effector travels in the direction of the target location—Alderbaran's method of solving the inverse kinematics uses the inverse Jacobian matrix to iteratively modify joint angles. Furthermore, to prevent jerky motions, smooth transitions between points are often required in robotic motion. This is accomplished by using interpolation methods like spline and linear interpolation. Simple, brief motions can benefit from the direct path that linear interpolation offers between two positions. Spline interpolation, especially when employing Bézier curves, which are mathematical constructions that are used to produce beautiful and flowing curves between specified control points, is recommended for movements that are more intricate or linked. Enhancing the capability and integration of these robotic devices is Alderbaran's NAOqi control framework as shown in Exhibit 1. It can function smoothly in a variety of programming environments by supporting cross-platform and cross-

language communication capabilities. For the robot to move precisely and in accordance with the desired programming, the framework's effective management of its sensors and actuators is essential. Put simply, the SPL and RoboCup offer a dynamic testing ground for advanced robotics and AI research, with an emphasis on the creation of autonomous humanoid robots that are able to execute intricate football movements. To progress robotic capabilities and meet the challenging objectives of RoboCup, mechanical design must be integrated with advanced control algorithms, as those made possible by the NAOqi framework.

Exhibit 1. An Illustration of a Network That Shows the Naoqi Communication System.



Related Works

The use of RL to optimize walking engines and penalty kicks is a major focus of the exploration of current kicking engine implementations for the Alderbaran Nao robot. Understanding which approaches could be most useful for improving robotic kicking actions in the RoboCup SPL is crucial, and our analysis will help. Determining the required parameters and features for the RL tasks described later in the study requires an understanding of the techniques employed by different RoboCup SPL teams. A trajectory-based technique that dynamically modifies a kick's direction to account for changes in the surrounding environment was employed by IPAB | InfWeb. This approach, which is also applicable to their trajectory-based walking motions, entails feeding a set of dynamic points into a motion engine that characterizes the joint angles of the kick. Inverse kinematics is then used to determine the other joint angles. With the use of cubic Bezier curves (Sa et al., 2016) enables real-time adjustments to the kick trajectory, guaranteeing fluid motion. (Shah et al., 2023) breaks the kicking motion down into smaller movements that are universal to all kicks-lifting the leg, kicking it back, coming up to the ball, following through, going back to the starting position, and putting the foot down. Daghottra et al. (2021) discusses Bezier curves that are smoothly connected by guaranteeing that the control points connecting each of these sub-motions are continuously differentiable characterize each of these sub-motions such that the robot is kept from falling during the kicking motion by using a closed-loop PID controller with sensory feedback and a balance module that uses Centre of Mass (COM) stabilization in addition to trajectory updates. Using input from the robot's gyroscope, the PID controller corrects for outside disturbances while the COM stabilization corrects for any imbalance that may occur during the kick.

Robotics effectiveness is determined by how well it can approach and kick a ball that is positioned at random with the least amount of angle deviation. The significance of precisely optimized kicking motions is demonstrated by (Koenig et al., 2004), which significantly outperforms the prior methods in terms of kicking distance (Kamuni et al., 2024). A different strategy was used by (Dodda et al., 2024), who chose the kick type after taking the opponent's position into account. The engine of their kicking motion is able to adapt to the ball's position in relation to the foot, allowing for both walking and static standing kicks. Pre-kick positioning is emphasized in this method to maximize the impact direction and distance, with less focus on dynamic modifications during the kick. Kumar et al. (2024) devised an omnidirectional stationary kick that adjusts its start and end points dynamically in response to the ball's position in relation to the foot during the propulsive phase. The lateral movement of the foot up to 120 mm from the center of the ball enables angled kicks with this technique, offering a flexible kicking strategy for a range of game scenarios. To summarize, the various methods used to create kicking engines for humanoid robots emphasize the significance of dynamic motion control, balance stabilization, and pre-kick changes for strategy. Regarding the application of RL to improve robotic performance in competitive contexts (Soni, Alla, Dodda, & Volikatla, 2024) such as RoboCup, each team's tactics offer insightful information. To enhance balance, accuracy, distance, and speed of execution in dynamic, real-world scenarios, these discoveries lay the groundwork for future development of RL applications in robotic kicking actions. By improving the interaction algorithms between robots and their operational settings, the study of these approaches increases not only the specialized abilities of robots that play football but also makes a broader contribution to the area of robotics.

Pre-Experiment

Preliminary testing is necessary to validate the results from relevant research before putting the RL algorithm into practice. In order to assure accuracy and reliability, any programs created for the robot should first be checked in a simulator. This will save time and money by minimizing the need for expensive in-person testing. Because modern robots are expensive, testing robotic behaviors can be costly both in terms of time and setup as well as cost because of the possibility of damage from mechanical stress and falls. The joints of robots, especially those utilized in research projects such as the RoboCup have motors that can overheat and wear out under high loads and repetitive motions. Because it reduces the dangers to actual hardware, simulation becomes an appealing first step. Without the need to physically do these tests, simulators use advanced physics engines to recreate the physical qualities of the real world. This makes it possible to evaluate robotic activities, such as kicking motions, quickly. Because each iteration requires a significant amount of setup and tear-down time, this effective arrangement allows for the rapid iteration of tests, hundreds or even thousands of times, which would not be possible in a real-world environment. Nonetheless, there are certain difficulties with using simulators. Reliable simulation results necessitate extremely accurate physics calculations and meticulous robot and environment modelling. To accurately reflect real-world circumstances, parameters like mass, gravity, and collision impacts must be carefully set. And yet, with proper implementation, simulations can shield robots from harm while yielding insights that are on par with those from real-world experiments. Unpredictable environmental factors like surface differences or mechanical slippage, as well as damage from falls and overheating motors, might cause problems during physical testing and distort the results. SimSpark, a simulator utilized in the RoboCup simulated league, is one example of an existing simulator that was judged to be inadequate for our particular requirements as of the time of writing. An essential component for smooth transitions between simulated and real-world testing is the NAOqi framework, which is not integrated into SimSpark. This framework is used to manage the Nao robots. SimSpark, on the other hand, makes use of HTTP protocols, which don't function with the frameworks we have. Although NAOsim, another simulator from Alderbaran, makes use of the NAOqi architecture, it is unable to precisely measure or script environmental interactions, a feature that is crucial for testing RL. NAOsim's limitations prevent it from being used for the extensive testing necessary for this research, even though it is appropriate for basic movement tests and balance control. Eventually, real-world testing is still necessary to fine-tune behaviors and make sure algorithms function as intended in real, dynamic environments. Simulations are helpful for the preliminary testing stages of this process, as they guarantee the safety and initial viability of robot motions. The objective of this research is to create a strong RL framework that can accurately model and enhance the kicking motions of Nao robots in the RoboCup competition environment. This will address the effectiveness of learning algorithms as well as the robots' physical performance through the careful application of both simulated and real-world testing.

Post-Experiment

Optimizing robotic movements for many purposes at once has been a central difficulty in the fast-developing field of robotics. Previous research has focused on improving the precision or balance of robotic kicking motions. This study presents a more comprehensive method to optimize kicking motions based on a variety of parameters, including distance, angular accuracy, execution time, and stability, utilizing off-policy temporal difference Q-learning. Therefore, this implementation aims to apply these policies to directed kicks at a 45° angle and develops optimal policies that can establish effective kicking movements across five various distances (1, 2, 3, 4 meters, and as far as possible) straight ahead at a 0° bearing. For the benefit of the kicking motions (Barrett et al., 2024), the best policies found using this RL technique will be included into a kicking engine. Based on characteristics including distance, ball displacement in relation to the robot, and the intended kick angle, this engine will be able to choose and modify kicking actions. In order to do this, Alderbaran's NAOqi control framework is used to design the software required to apply the RL algorithm and control the robot. This involves making use of the Python programming language, which although having a slower runtime than C++, is useful for managing state and value tables due to its ease of handling list manipulations. Bézier interpolation and inverse kinematics are used in this setup to execute motion, and the Cartesian control API from the NAOqi Python SDK is used to move end-effectors around Cartesian space. Therefore, kicking motion can be categorized into five main phases, which are shown by key-frame positions shifting weight, elevating the kicking leg, pushing it backwards, driving it forward to kick, and follow-through. To calculate the necessary velocity for each movement phase, the motion between these frames is interpolated. Motion between these frames is interpolated using a combination of linear and curved interpolations to optimize motor velocities for smoother motion determining the required velocity for each movement phase. Reduced search space size for the RL issue is a major component of this technology. To do this, count the DoF that are essential and eliminate any that have no bearing on the kick, like foot roll. While taking into account mechanical limitations such as Nao's leg thickness

and hip joint configuration, emphasis is made on the foot's pitch to improve accuracy and the use of the foot's top for contact to guarantee the ball travels in the right direction. By using a state-space that incorporates coordinates, joint rotations, and execution time, the RL method uses each state to represent a key-frame of the kicking leg. The movement instructions required to change states within set time intervals define the action-space. A controlled search environment that facilitates faster learning and optimization is created with the help of this methodical technique. In order to enable RL to effectively traverse the numerous possible state and action combinations, an adaptation of the Q-learning algorithm is used in this instance. To limit misuse and potential damage to the robot's actuators, the program uses an ε-greedy policy to balance exploration and exploitation. It refines policies within a finite number of episodes. The development of a dynamic kicking engine is the last implementation detail. This module selects and adjusts the proper kicking motions based on input factors such as the intended kick angle and the ball's position in relation to the robot. This flexibility is critical for real-world settings, like RoboCups, where prompt and precise reactions are required. This all-encompassing method works to improve robotic kicking skill, but it also advances robotics by showcasing the practical use of RL in everyday tasks.

Result Analysis

In order to minimize environmental differences among tests, this section describes the experimental setup created to evaluate the Q-learning algorithm on a limited state-space. Only the backswing distance and reward function are changed between tests in each of the two experiments, each consisting of 100 episodes. Finding the longest kicking distance is the goal of the first set of experiments. Three different tests with backswings of 2 cm, 5 cm, and 8 cm are conducted using the long-distance reward-function. To confirm if additional experience results in a better greedy policy, the most efficient backswing distance from these first experiments is subsequently used in a longer 200-episode experiment. Although this setup may offer differences from the carpet surface and line tape, the lengthier test is carried out diagonally across the pitch in an attempt to potentially leverage wider distances. However, using the best backswing found in the maximum distance test, set distance reward-functions are used in the second trial type to optimize policies for certain distances. By specifying the strategy type for the learning algorithm, several reward-functions can be changed. The ankle has been rotated in the keyframes, and the *minZ* value has been adjusted to 5.5 cm to prevent the toe edge from contacting the ground. The aim of this arrangement is the same as the maximum distance tests, to travel the farthest distance feasible. We test the final key-frames for each kick 20 times to find the standard deviation and variances once optimal policies for all distances are determined. This gives an idea of the ultimate policy' effectiveness. It is anticipated that there will be more variations between kicks in a genuine RoboCup scenario if kicks are taken from the goalkeeper's position and aim down the pitch over line marking tape. Each strategy is tested in the last kicking module by positioning the ball at varied distances and angles from the center of the kicking foot—for every type of distance, straight kicks with ball displacements of ±1 and ±2 cm are used, and angled kicks (20°, 30°, 45°) at all ranges, both with and without displacement.

We'll assess how well the kicking module performs in terms of on-time achievement of the necessary distances, angles, and stability. Using 11 DoF for the kicking action, the tests use a 25-DoF H25 v3 Alderbaran Nao robot. Manual measurements are used to ensure uniformity, including shot angle and ball travel. The robot can move freely during tests because it is protected from falls by a harness. In order to ensure uniformity, a template is used to place the robot, guaranteeing the same beginning position for every test. Basic trigonometry is used to estimate the angular accuracy and measure the kicking distance using a tape that runs the length of the pitch. The kick's speed is measured as the total of the interpolation times for each phase. Despite the robot's symmetry, in order to prevent motor disparities, just the left leg is used in the early tests. Each policy is tried on the right leg to verify consistent outcomes once the best kicks have been determined.

Maximum Distance. A comparison of the kicking distance across episodes is shown in Exhibit 2, which shows how the RL agent explored a variety of policies. With the policies starting to take advantage of more states after episode 82, there is a discernible trend of longer distances, yet lower-scoring episodes are still sometimes seen. It is evident from the tests that the greedy rules that are ultimately used in both studies have comparable follow-through and backswing key-frames with respect to interpolation times and height. Different approaches are taken to make contact with the ball, Policy 1 makes contact 1.5 cm below the ball's center line, while Policy 2 makes contact 1.5 cm above it, both staying equally distance from the center. These policies ensure an unchanged bearing, and a minimum execution time, all of which are good matches for the optimization criteria. The maximum distance reached, in spite of these optimizations, was 482 cm, exceeding the original kick by 76.4 cm but falling short of the benchmark distance of 523 cm. Only three of the tests saw the robot experience negative rewards because of bearing changes; otherwise, it remained stable and did not tumble. According to test 1's heat map (Exhibit 3), the states chosen in greedy policy 1

are the ones with the highest values during the first two time-steps. A difference in optimal state values between kick phases can be seen in the preference for a height of 4 cm during the follow-through phase (time-step 3) as opposed to the 5 cm height selected by the greedy policy. An intriguing pattern is found by analyzing the policies' backswing and follow-through. When it comes to follow-through, both policies engage the ball at a height of 4 cm above the ground, which is marginally higher than the ball's center. Policy 1 raises the height by 1 cm, while Policy 2 lowers it by 1 cm. Other elements of Policy 2 include a longer interpolation time of 0.2 seconds and a lower backswing that is aligned with the ball. Episodes 87 and 88, in particular, investigated a faster backswing of 0.1 seconds, which resulted in a negative reward because of a shift in bearing. This state is indicated in dark blue on the heat map (Exhibit 4). These instances, in which there were negative incentives, skew the average action-values shown in the heat map. Policies 1 and 2 produced the longest kicks in terms of distance, with backswings of 5 cm and 513 cm, respectively, being the largest. This is little better than the benchmark distance of 523 cm, averaging a maximum distance of 529.5 cm. The regulations also show that, although robot sometimes experiences negative rewards when using the fastest interpolation times, an interpolation time of 0.3 seconds is often effective. Robotic performance was stable throughout testing, with no instances of falling. This shows how well RL can be applied to optimize robotic kicking motions while maintaining stability and precision objectives.

Exhibit 2. Distances Obtained from the Greatest Distance Test with a 2-centimeter Backswing.

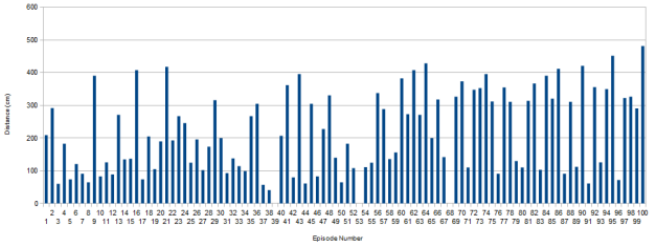


Exhibit 3. State-Value Table Heat Map for a Long Distance with a 2 cm Backswing.

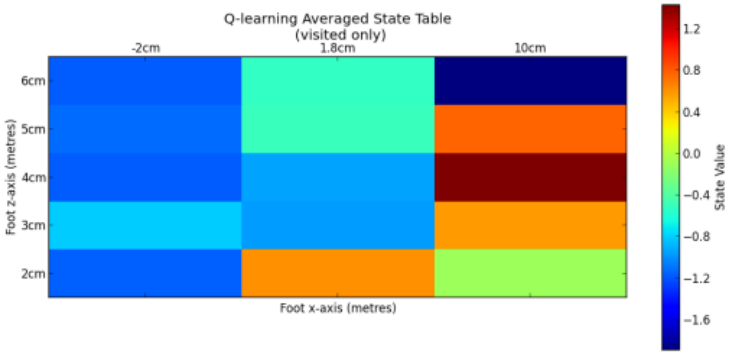
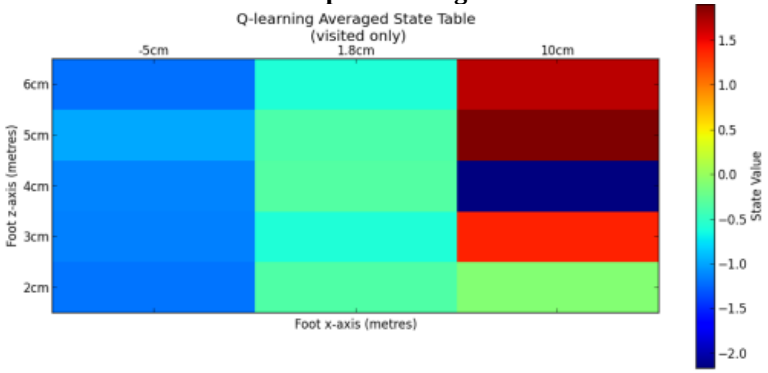


Exhibit 4. State-Value Table Heat Map for a Long Distance with a 5 cm Backswing.



Specific Distance. By episode 37, as the exploitation rate reaches 25%, the agent's behavior dramatically shifts, as shown in Exhibit 5. The majority of kicks resulting from this shift are low-distance, with only 12 episodes going over 150 cm and 5 over 200 cm. Notwithstanding these results, in 4 of the tests 1 instances, the agent reaches the minimum goal distance of 100cm. Regarding stability, the greedy policy in test one sustains it for eleven episodes, whereas in test two, it becomes stable for nine episodes. In Policy 2, the ball is given a small amount of backspin by being struck high and followed through low, which reduces the ball's rolling distance. But in Policy 1, the agent is penalized twice for failing to kick and three times for changing bearing during the backswing. This is because Policy 1 uses a curved motion, just like in earlier testing. In comparison to the initial kick time of 0.8 seconds, both policies achieve optimization criteria by keeping the robot stable, avoiding bearing shifts, and achieving the specified distance quickly. Both policies are able to hit exactly 100 cm at least once. Because Policy 1 executes more quickly than the others, it turns out to be more effective. There is a clear preference for a follow-through height of 5 cm and a contact point height of 2 cm, as seen in the heatmap (Exhibit 6) from this test phase. All backswing phases receive low scores, which are indicative of the less effective execution times connected to short kicks. As the episodes go on, the agent gains enough state-space information by episode 71 to effectively exploit optimal policies, as shown in Exhibit 7. After episode 71, 19 more episodes have kicks in the top reward category of 20 cm; the greatest distance ever recorded was 199 cm by greedy policy 1. For 13 episodes in test one and 11 episodes in test two, the greedy policy shows performance consistency. The backswing strategies of the two policies are different, although they both show comparable trajectories and time interpolations. Both approaches ensure the robot's accuracy and stability during kicks, as shown by the heatmap in Exhibit 8, which verifies that after 100 episodes, all backswing states are rated equally. At 0.5 seconds less than its counterpart, Policy 1 completes motion more quickly. The values of the backswing state vary very little in heatmap, as in previous experiments. The states with the greatest average value of 2.5 across all tests thus far, the contact point at 3 cm and follow-through at 6 cm, stand out as the most valuable ones. This suggests a consistent approach to obtaining optimal kick performance by deliberate policy exploitation.

Exhibit 5. The Distances Obtained from the One-Meter Reward Function Test.

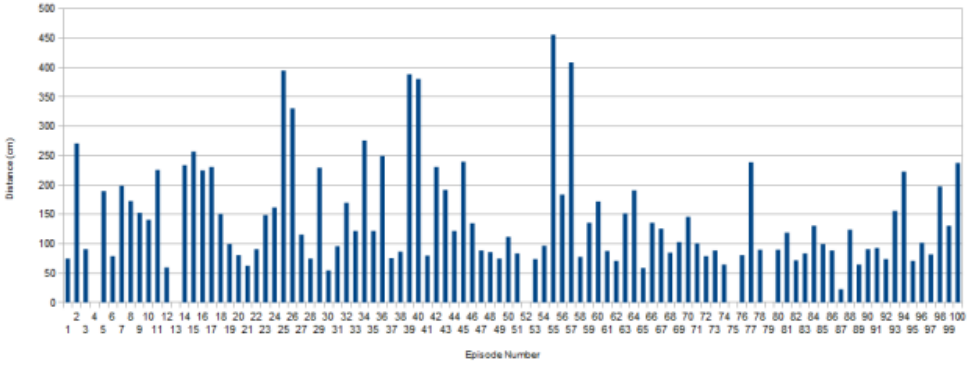


Exhibit 6. Heat Map of the State-Value Table at One Meter.

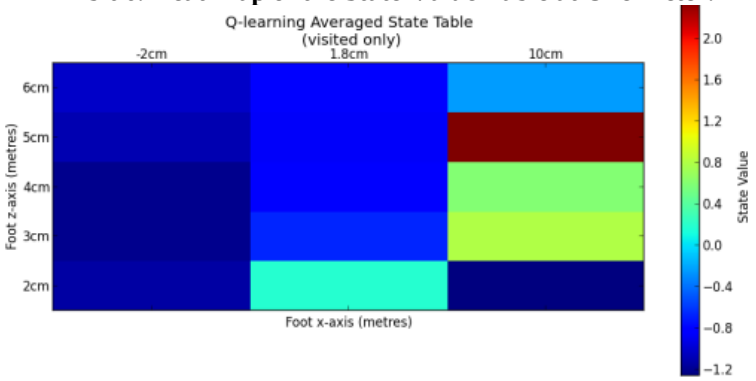


Exhibit 7. The Distances Obtained from the Two-Meter Reward Function Test.

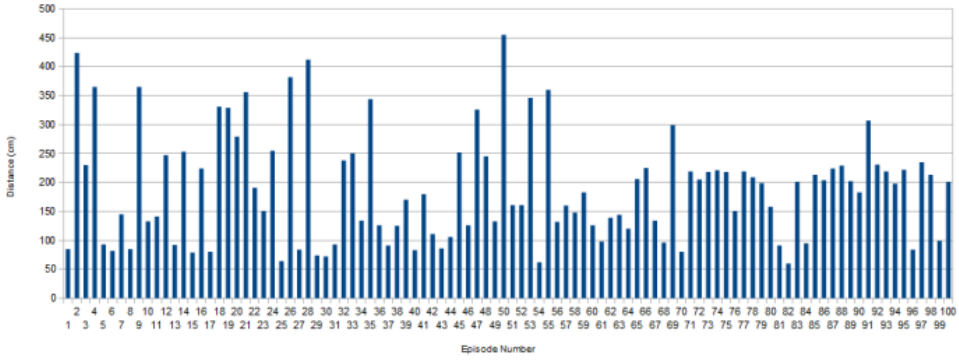
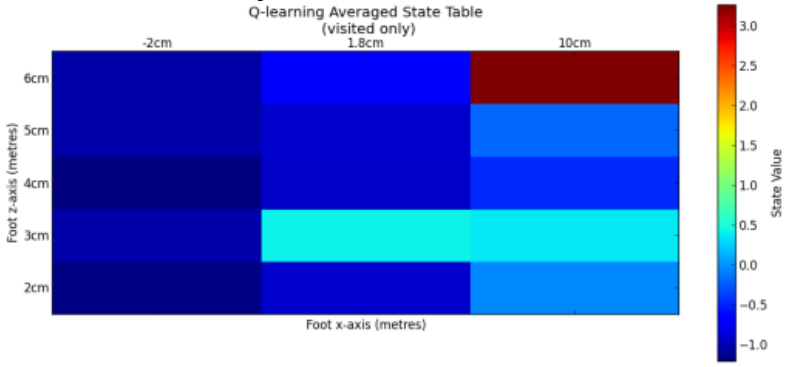
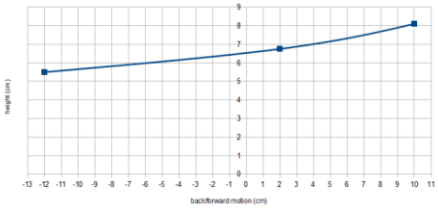


Exhibit 8. Heat Map of the State-Value Table at Two Meters.



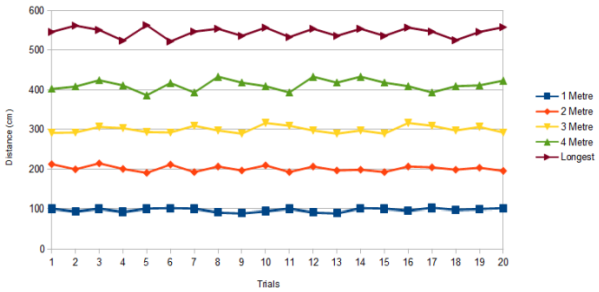
Ankle Rotation. Throughout a thorough trial, a soccer-playing robot presented several difficulties such as the complexity and constraints of its mechanical and software design. During the experiment, the robot fell an astounding 80 times, indicating a serious instability in its functioning. Furthermore, it experienced eight bearing adjustments and missed four kick attempts. The robot managed to carry out eight kicks, one of which travelled as far as 420 centimeters, in spite of this many failures. This outcome was attained by applying a particular greedy policy described in the trajectory displayed in Exhibit 9. The robot's ankle, which was designed to work for 0.6 seconds, was used in the kicking motion that was being examined. Other maximum distance greedy policies found in the study had execution times that were significantly less than this one. It was discovered that mechanical problems resulted from a reduction in the ankle's longer duration of motion, which made it crucial. To be more precise, the robot's hip motor tended to lock as the execution time was lowered. As a result of this defect, the knee and foot joints unintentionally extended backward, which was the main reason for most of the experiment's falls. The robot's foot height increased with each forward phase of its kick, as shown by the analysis of the trajectory curve in Exhibit 9. The direction and power of the kicks were immediately affected by this change in the kicking method, which made it extremely important. The ball was launched straight forward in the desired direction in the cases when the robot successfully executed its kick without falling. This showed that the robot was able to perform precise and powerful kicks when its mechanical motions were appropriately synchronized in accordance with the greedy policy. The intricate interplay of the robot's mechanical configuration, movement execution timing, and programming regulations is highlighted by these results. Improved stability and dependability of the hip and ankle motors are two areas that could benefit from attention, as seen by the frequent falls and mechanical lock-ups. A more stable robotic performance could result from improving these factors, which would lower the chance of falls and raise the kicks' overall success rate. The results of the experiment further imply that in order to maximize performance, precise adjustments must be made to the interpolation times and the kicking strategy's movements. To find more durable and dependable kicking motions, future research could investigate modifications in these factors. Further development of algorithms for error handling and real-time adjustments may also assist alleviate some of the problems related to motor lock-ups and bearing changes. All things considered, the experiment gave important new insights into the difficulties involved in building and programming a robot for a physically demanding activity like football. It emphasized the careful balancing act needed to accomplish desired performance outcomes between mechanical design, programming accuracy, and timing of execution.

Exhibit 9. Foot motion trajectory with ankle rotation for greatest distance.



Optimized Kicks. The performance of greedy policies was examined in five different distance tests through a thorough investigation of robotic kicking motions. The effectiveness of each of these strategies was tested by having 20 consecutive kicks made; the results were tallied and graphically displayed in Exhibit 10. Critical insights into the efficacy and unpredictability of the kicking motions at various predetermined distances were offered by this data. One important finding from the research was the correlation between the goal distances for each test and the standard deviation (σ) of the kick distances. The results showed that the kicks' standard deviation rose with the desired distance. According to this pattern, longer kicks are generally less consistent than shorter ones, probably because regulating more forceful and complicated movements comes with more mechanical and computational difficulties. These results have important ramifications for forecasting how reliably the robot will function in real-world situations. The data allowed for an assessment of the worst-case scenario accuracy for each kick, since normally distributed samples usually deviate by 2σ from the mean about 95% of the time. In particular, it was determined that under the worst circumstances, a kick would land within 27.68 cm of the intended distance. For the purpose of strategic planning in applications such as robotic football, this metric offers a quantitative estimate of the expected accuracy and reliability of the robotic kicks. To improve the design and programming of robots that must operate in environments demanding great precision and consistency, it is imperative to comprehend these dynamics. Engineers and developers can modify the robot's design to account for the observed inconsistencies by improving the stability of the drive systems or optimizing the control algorithms, based on the observation of a rise in variance with distance. This analysis also emphasizes how crucial it is to do thorough testing and calibration when developing robotic systems since consistency in performance is crucial. Reducing variance and improving the overall accuracy of the robot's actions are the main goals of changes that can be strategically applied by identifying the patterns and limitations that systematic testing reveals. This work advances our understanding of the physical and computational constraints that robotic systems inevitably face in addition to helping to improve the technical details and operational algorithms of the robots. These kinds of insights are very helpful for robotics research and development, especially for applications where accuracy and dependability are critical.

Exhibit 10. Tested Twenty Times to Determine the Best Kicking Actions for Each Distance.



Conclusion and Future Works

As the study came to an end, Alderbaran unveiled Webots for Nao, a new simulator that improved environmental manipulation and NAOqi integration, but it wasn't adequate to be used for this particular study. By providing more intricate surroundings and multiple simulation iterations, future research should make use of this simulator in order to potentially achieve the learning agent's full potential. A RL agent that can handle different states, time-steps, and interpolation times was put into practice by this study. In the future, this might be improved by applying value function approximation to lessen the requirement for large amounts of experience data and by generalizing similar state-action combinations using methods such as gradient descent, which would increase the learning scope without requiring repeated state visits. In order to improve flexibility in dynamic contexts like RoboCup, more research could optimize various kicking patterns, such as side-sweeping or walking kicks. This could be accomplished by employing a

collection of policies to enable kicks at any walking phase. Furthermore, enhancing the kicking module to update key-frames dynamically in real-time and maybe increasing the reach of the kicking leg by modifying the robot's weight distribution could improve performance. Adding a balancing module could help improve stability during kicks, preventing the momentum-induced instability caused by the Nao robot's smooth soles and quick actuators that are currently in use. The study's findings highlight the effectiveness of RL in identifying the best course of action for intricate, multi-objective robotic tasks. It also shows that RL may be used to create advanced, dynamic robotic movements without requiring large amounts of pre-modeled surroundings. This effort paves the way for testing and more advanced applications using the new Webots for the Nao simulator.

References

Alla, S., Soltanisehat, L., Tatar, U., & Keskin, O. (2018). Blockchain technology in electronic healthcare systems. In *IIE Annual Conference. Proceedings* (pp. 901-906). Institute of Industrial and Systems Engineers (IISE).

Alla, S., Soltanisehat, L., & Taylor, A. (2018, July). A Comparative Study of Various AI Based Breast Cancer DetectionTechniques. In *IX International Conference on Complex Systems* (p. 213).

Alla, S., & Pazos, P. (2019). Healthcare robotics: Key factors that impact robot adoption in healthcare.

Alla, S. (2019). A Statistical Analysis of Surgeons' Preference on Robot-Assisted Surgeries. In *IIE Annual Conference. Proceedings* (pp. 1385-1390). Institute of Industrial and Systems Engineers (IISE).

Daghottra, A., & Jain, D. (2021). From humans to robots: Machine learning for healthcare. *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, 705-714.

Dodda, S., Kumar, A., Kamuni, N., & Ayyalasomayajula, M. M. T. (2024). Exploring Strategies for Privacy-Preserving Machine Learning in Distributed Environments. *Authorea Preprints*.

Kamuni, N., Cruz, I. G. A., Jaipalreddy, Y., Kumar, R., & Pandey, V. K. (2024). Fuzzy Intrusion Detection Method and Zero-Knowledge Authentication for Internet of Things Networks. *International Journal of Intelligent Systems and Applications in Engineering*, *12*(16s), 289-296.

Kamuni, N., Shah, H., Chintala, S., Kunchakuri, N., & Alla, S. (2024, February). Enhancing End-to-End Multi-Task Dialogue Systems: A Study on Intrinsic Motivation Reinforcement Learning Algorithms for Improved Training and Adaptability. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)* (pp. 335-340). IEEE.

Kamuni, N., Chintala, S., Kunchakuri, N., Narasimharaju, J. S. A., & Kumar, V. (2024, February). Advancing Audio Fingerprinting Accuracy with AI and ML: Addressing Background Noise and Distortion Challenges. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)* (pp. 341-345). IEEE.

Kamuni, N., Jindal, M., Soni, A., Mallreddy, S. R., & Macha, S. C. (2024). A Novel Audio Representation for Music Genre Identification in MIR. *arXiv preprint arXiv:2404.01058*.

Kashyap, G. S., Mahajan, D., Phukan, O. C., Kumar, A., Brownlee, A. E., & Gao, J. (2023). From Simulations to Reality: Enhancing Multi-Robot Exploration for Urban Search and Rescue. *arXiv preprint arXiv:2311.16958*.

Koenig, N., & Howard, A. (2004, September). Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ international conference on intelligent robots and systems (IROS)(IEEE Cat. No. 04CH37566)* (Vol. 3, pp. 2149-2154). Ieee.

Kumar, A., Dodda, S., Kamuni, N., & Vuppalapati, V. S. M. (2024). The Emotional Impact of Game Duration: A Framework for Understanding Player Emotions in Extended Gameplay Sessions. *arXiv preprint arXiv:2404.00526*.

Kumar, A., Dodda, S., Kamuni, N., & Arora, R. K. (2024). Unveiling the Impact of Macroeconomic Policies: A Double Machine Learning Approach to Analyzing Interest Rate Effects on Financial Markets. *arXiv preprint arXiv:2404.07225*.

Lakshminarayanan, V., Ravikumar, A., Sriraman, H., Alla, S., & Chattu, V. K. (2023). Health care equity through intelligent edge computing and augmented reality/virtual reality: a systematic review. *Journal of Multidisciplinary Healthcare*, 2839-2859.

Marwah, N., Singh, V. K., Kashyap, G. S., & Wazir, S. (2023). An analysis of the robustness of UAV agriculture field coverage using multi-agent reinforcement learning. *International Journal of Information Technology*, *15*(4), 2317-2327.

Shah, H., & Kamuni, N. (2023, December). DesignSystemsJS-Building a Design Systems API for aiding standardization and AI integration. In *2023 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA)* (pp. 83-89). IEEE.

Soni, A., Alla, S., Dodda, S., & Volikatla, H. (2024). Advancing Household Robotics: Deep Interactive Reinforcement Learning for Efficient Training and Enhanced Performance. *arXiv preprint arXiv:2405.18687*.