

Reinforcement Learning for Robot Navigation

Kumar Suyash¹, Pratyush Kumar², Ayush Kumar Dubey³, Shubh Tiwari⁴, Abhishek Tiwari⁵, Dr. Pavithra G⁶ and Dr. T.C.Manjunath⁷

¹⁻⁵Final year BE (ECE) Students, VII Sem, Electronics & Communication Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka

⁶Associate Professor, Dept. of Electronics & Communication Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka

⁷Professor & HOD, Dept. of Electronics & Communication Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka

Email: tcmanju@iitbombay.org

Abstract— This paper undertakes a comprehensive exploration of the application of Reinforcement Learning (RL) in robotic systems, with a specific focus on the navigation tasks carried out in intricate environments. The primary objective is to elevate the adaptability, efficiency, and generalization capabilities of robots. The study delivers an insightful overview of RL concepts, accentuating their relevance in the context of robot navigation. It systematically evaluates various RL algorithms, with a particular emphasis on Deep Reinforcement Learning (DRL). The analysis incorporates crucial considerations such as state representation, action space, and reward shaping. In addition to scrutinizing RL algorithms, the paper delves into the integration of sensor data, mapping techniques, and Simultaneous Localization and Mapping (SLAM) within RL frameworks, aiming to enhance the robot's perception capabilities. The discussion extends to the adaptation of RL methodologies to address real-world challenges, including dynamic obstacles and unknown terrains. The paper substantiates its findings through the presentation of case studies and experiments that exemplify practical applications of RL in robotic navigation. The conclusion of the paper encapsulates a thoughtful discussion on existing challenges, potential future research directions, and anticipated advancements in the field. The overarching goal is to provide a comprehensive resource for researchers and practitioners interested in harnessing RL to augment autonomous robot navigation within complex and dynamic environments. By offering a thorough examination of RL applications and addressing practical challenges, this paper contributes significantly to the evolving landscape of robotics research and technology.

Index Terms— Autonomous navigation; Collision avoidance; Reinforcement learning; Mobile robots.

I. INTRODUCTION

Autonomous navigation of robots in complex and dynamic environments poses a significant challenge in the field of robotics. As the demand for intelligent and adaptive robotic systems continues to grow, the exploration of advanced techniques becomes imperative. Reinforcement learning (RL), a subfield of machine learning, has

of robotics. As the demand for intelligent and adaptive robotic systems continues to grow, the exploration of advanced techniques becomes imperative. Reinforcement learning (RL), a subfield of machine learning, has emerged as a promising paradigm for enabling robots to autonomously navigate through intricate surroundings. The ability of RL algorithms to learn from interactions with the environment, adapt to changing conditions, and optimize decision-making processes aligns seamlessly with the demands of robot navigation. This paper delves into the application of reinforcement learning in addressing the multifaceted challenges associated with robot navigation. Navigating in real-world environments requires robots to possess not only the capability to perceive and understand their surroundings but also the adaptability to handle dynamic obstacles, unknown terrains, and unforeseen changes in the environment. RL, with its ability to learn optimal policies through trial-and-error interactions, offers a compelling approach to imbue robots with the intelligence needed for efficient and flexible navigation.

II. LITERATURE REVIEW

Reinforcement learning: an introduction - The significantly expanded and updated new edition of a widely used text on reinforcement learning, one of the most active research areas in artificial intelligence. Reinforcement learning, one of the most active research areas in artificial intelligence, is a computational approach to learning whereby an agent tries to maximize the total amount of reward it receives while interacting with a complex, uncertain environment. In Reinforcement Learning

Efficient off-policy meta-reinforcement learning via probabilistic context variables - Deep reinforcement learning algorithms require large amounts of experience to learn an individual task. While meta-reinforcement learning (meta-RL) algorithms can enable agents to learn new skills from small amounts of experience, several major challenges preclude their practicality. Current methods rely heavily on on-policy experience, limiting their sample efficiency.

Learning image-conditioned dynamics models for control of under-actuated legged millirobots - This study explores the potential of millirobots, specifically legged variants, as a promising and cost-effective robotic platform for diverse applications. While these miniature robots offer increased mobility in complex environments and improved obstacle scaling, their control poses significant challenges. Traditional hand-engineered controllers face limitations in managing dynamic maneuvers and complex terrains. In response, we present an innovative approach utilizing learned neural network models for the control of real-world legged millirobots.

Reinforcement learning in robotics: a survey - Reinforcement learning offers to robotics a framework and set of tools for the design of sophisticated and hard-to-engineer behaviors. Conversely, the challenges of robotic problems provide both inspiration, impact, and validation for developments in reinforcement learning. The relationship between disciplines has sufficient promise to be likened to that between physics and mathematics.

Guided cost learning: deep inverse optimal control via policy optimization - This study applies reinforcement learning to achieve complex behaviors in high-dimensional robotic systems, focusing on torque control. Overcoming challenges in defining an effective cost function, the authors use Inverse Optimal Control (IOC). Their algorithm learns arbitrary nonlinear cost functions without detailed feature engineering and provides an efficient sample-based approximation for MaxEnt IOC. Evaluation in simulated tasks and real-world robotic manipulation shows substantial improvements in both task complexity and sample efficiency compared to prior methods.

Emergence of locomotion behaviours in rich environments - This paper explores the reinforcement learning paradigm's potential to learn complex behaviors directly from simple reward signals. While it's common in practice to hand-design reward functions for specific solutions, this study investigates the impact of a rich environment on promoting the learning of complex behavior. The approach involves training agents in diverse environmental contexts, leading to the emergence of robust behaviors that perform well across various tasks. The focus is on locomotion, a behavior known for its sensitivity to reward choices.

Smart Element Aware Gate Controller For Intelligent Wheeled Robot Navigation - This paper introduces a novel approach for directing a wheeled robot through an unknown and dynamic environment filled with physical barriers. Emphasizing the achievement of an optimal or near-optimal path while avoiding obstacles, the proposed solution features a modified neuro-controller mechanism. This mechanism is built upon a Modified Elman Neural Network (MENN) incorporating an Effective Element Aware Gate (MEEG) as the neuro-controller [1].

Combining finite state machine and decision-making tools for adaptable robot behavior - Modeling robot behavior is a common task in robot software development. However, its difficulty grows exponentially along with system complexity. To facilitate the development of a modular, rather than monolithic, behavior system, proper software

tools need to be introduced. This paper proposes combination of a well-known finite state machine and a custom decision-making tool for implementing adaptive robot behaviors [2].

A review on indoor human aware autonomous mobile robot navigation through a dynamic environment survey of different path planning algorithm and methods - In complex indoor applications with numerous bedrooms, hallways, and doorways containing both static and dynamic obstacles, the disintegration of the environment into smaller regions significantly influences the quality of adaptive path planning. Various algorithms address challenges like narrow passages and multiple regions, emphasizing optimal solutions. For an autonomous mobile robot, achieving a balance between potential, stability, and robustness is crucial for effective adaptive path planning and navigation in dynamic environments [3].

Model-agnostic meta-learning for fast adaptation of deep networks - We propose an algorithm for meta-learning that is model-agnostic, in the sense that it is compatible with any model trained with gradient descent and applicable to a variety of different learning problems, including classification, regression, and reinforcement learning. The goal of meta-learning is to train a model on a variety of learning tasks, such that it can solve new learning tasks using only a small number of training samples [4].

Navigating occluded intersections with autonomous vehicles using deep reinforcement learning - Providing an efficient strategy to navigate safely through unsignaled intersections is a difficult task that requires determining the intent of other drivers. We explore the effectiveness of Deep Reinforcement Learning to handle intersection problems. Using recent advances in Deep RL, we are able to learn policies that surpass the performance of a commonly-used heuristic approach in several metrics including task completion time and goal success rate and have limited ability to generalize [5].

Continuous deep q-learning with model-based acceleration - In this paper, we explore algorithms and representations to reduce the sample complexity of deep reinforcement learning for continuous control tasks. We propose two complementary techniques for improving the efficiency of such algorithms. First, we derive a continuous variant of the Q-learning algorithm, which we call normalized advantage functions (NAF), as an alternative to the more commonly used policy gradient and actor-critic methods [6].

On generalized bellman equations and temporal-difference learning - We consider off-policy temporal-difference (TD) learning in discounted Markov decision processes, where the goal is to evaluate a policy in a model-free way by using observations of a state process generated without executing the policy. To curb the high variance issue in off-policy TD learning, we propose a new scheme of setting the λ -parameters of TD, based on generalized Bellman equations. Our scheme is to set λ according to the eligibility trace iterates calculated in TD, thereby easily keeping these traces in a desired bounded range [7].

Reinforcement learning and Markov decision processes - Situated in between supervised learning and unsupervised learning, the paradigm of reinforcement learning deals with learning in sequential decision making problems in which there is limited feedback. This text introduces the intuitions and concepts behind Markov decision processes and two classes of algorithms for computing optimal behaviors: reinforcement learning and dynamic programming. First the formal framework of Markov decision process is defined, accompanied by the definition of value functions and policies [8]. Figure 1 gives the flow chart & the block-diagram

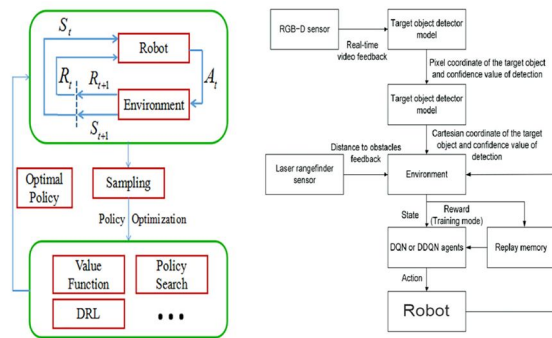


Figure 1. Flow chart & the block-diagram

III. WORKING OF THE ALGORITHM

This algorithm is a reinforcement learning approach for robot navigation. It involves initializing critic networks, replay memory, and setting parameters. During training, the robot interacts with the environment, selecting actions

based on a balance of exploration and exploitation. Q-values are updated using a learning rule, and networks are iteratively refined. The process includes sampling experiences, calculating loss, and updating both the critic and target networks. Exploration probability decreases over time. The algorithm is repeated for multiple episodes to enhance the robot's navigation capabilities in dynamic environments [9].

1. Initialization

- Initialize the critic network with weights θ .
- Initialize the target critic network with weights $\bar{\theta} = \theta$.
- Initialize replay memory D to capacity N.

2. Training Loop

- For each episode (Episode = 1 to K):
- Reset the environment.
- Get the initial state S_1 of the environment.
- Calculate the initial action A_1 .
- Set the current state S_t and action A_t to S_1 and A_1 .

3. Time-step Loop

- For each time-step ($t = 1$ to T):
- With probability ϵ , select A_t for S_t .
- Else, select $A_t = \text{argmax}_a q(S_t, A_t; \theta)$.
- Execute A_t .
- Observe R_{t+1} and S_{t+1} .
- Store $(S_t, A_t, R_{t+1}, S_{t+1})$ in D.
- Sample $(S_t, A_t, R_{t+1}, S_{t+1})$ from D.

4. Update Q-Values

- If $S_{t+1} = S_T$, set q DQN (or q DDQN) to R_T . Else, set:
- q DQN (s, a) = $R_{t+1} + \gamma \max_{a'} (q'(s', a'); \bar{\theta})$.
- q DDQN (s, a) = $R_{t+1} + \gamma q'(s', \text{argmax}_{a'} q(s', a'; \bar{\theta}); \bar{\theta})$.

5. Compute Loss

- Calculate $L = 1/M \sum_i (q \text{ DQN or } q \text{ DDQN} - q(s', a'; \bar{\theta}))^2$.

6. Update Critic Networks

- Update θ based on the computed loss.
- Update $\bar{\theta}$ every 500 time-steps.

7. Update Exploration Probability

- Update ϵ .

8. End of Training Loop

- End For (Episode loop).
- End For (Training loop).

IV. METHODOLOGY ADOPTED

The methodology for applying reinforcement learning to robot navigation in simpler terms [10]

1. *Understand the Problem:* Clearly define what you want the robot to achieve in its environment and figure out the challenges it might face.
2. *Set Up the Environment:* Create the physical or virtual space where the robot will move around. Install sensors (like eyes for the robot) and make sure the robot can move (using wheels or similar).
3. *Describe What the Robot Sees:* Decide how the robot will understand what's happening around it. This involves turning raw sensor data (what the robot 'sees') into a format that a computer can use.
4. *Decide What the Robot Can Do:* Define the actions the robot can take, such as moving forward, turning, or stopping.

5. *Reward the Robot*: Create a system to tell the robot when it's doing a good job (positive rewards) or when it's doing something wrong (negative rewards).
6. *Pick a Learning Method*: Choose a way for the robot to learn from its experiences. There are different methods, like learning from past experiences (Q-learning), learning from trial and error (Deep Q Networks), or learning by adjusting its actions (Policy Gradient methods).
7. *Connect with Mapping and Understanding Systems*: Help the robot understand where it is and what's around it by connecting the learning method with mapping and simultaneous localization and mapping (SLAM) systems [11].
8. *Teach the Robot*: Let the robot explore its environment and learn from its experiences. Record what it sees,

V. WORKING OF THE PROTOTYPE

1. *Robot Setup*: A physical or simulated robot is equipped with sensors to perceive its surroundings and wheels for movement [12], where the result is shown in the Fig. 2.
2. *State Representation*: The robot captures data from its sensors, such as images or distance measurements, representing the current state. This data is preprocessed to create a suitable input for the reinforcement learning algorithm [13].
3. *Action Space*: Actions are defined for the robot, such as moving forward, turning left or right, or stopping. These actions correspond to the robot's control mechanisms [14].
4. *Reward Mechanism*: A reward function is designed to provide feedback based on the robot's actions. For example, the robot might receive positive rewards for successfully avoiding obstacles or reaching a target location and negative rewards for collisions [15].
5. *Q-learning Algorithm*: The Q-learning algorithm is implemented to guide the robot's decision-making. The Q-values are updated based on the observed rewards and the transition to new states [16].
6. *Mapping and SLAM Integration*: The robot integrates mapping and SLAM techniques to enhance its understanding of the environment. This information is incorporated into the state representation to improve navigation decisions [17].
7. *Training Phase*: The robot goes through a training phase where it explores the environment, takes actions, and receives feedback. During this phase, the Q-learning algorithm learns optimal policies for navigation [18].
8. *Adaptation to Changes*: The robot is designed to adapt to changes in the environment. For example, if new obstacles appear or the layout of the space changes, the reinforcement learning model adjusts its policies accordingly [19].
9. *Testing and Evaluation*: The robot is tested in various scenarios to evaluate its navigation performance. This includes assessing how well it avoids obstacles, reaches goals, and adapts to changes in the environment [20].
10. *Fine-Tuning and Optimization*: Based on the evaluation results, the prototype may undergo fine-tuning. This involves adjusting parameters, refining the reward function, or optimizing the learning algorithm to enhance navigation performance [21].
11. *Real-world Deployment*: Once the prototype demonstrates satisfactory navigation capabilities, it can be deployed in real-world scenarios or more complex environments [22].

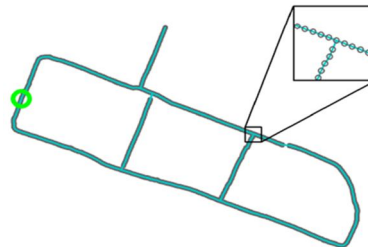


Figure 2. Path followed to reach the destination by the robot

VI. CONCLUSIONS

In conclusion, this paper has explored and presented a reinforcement learning approach tailored for robot navigation, demonstrating its efficacy through a single traversal of the environment. The contributions of this

work, including the implementation of an interactive replay buffer technique, the integration of stochastic observations into the training environment, and the use of pre-trained visual features, collectively contribute to the success of the proposed approach. The interactive replay buffer technique allows for the generation of extensive training experiences from real-world data, a crucial aspect for effective training of model-free systems. The incorporation of stochastic observations and the utilization of pre-trained visual features significantly enhance the performance of the system on both training and validation environments. Notably, these contributions prove instrumental in achieving reliable zero-shot transfer to environmental variations not encountered during the training phase.

REFERENCES

- [1] Iure Rosa L. Oliveira, Alexandre S. Brandão, "Deep Reinforcement Learning for Mapless Robot Navigation Systems", 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE), pp.331-336, 2023.
- [2] Iure Rosa L. Oliveira, Alexandre S. Brandão, "Deep Reinforcement Learning for Mapless Robot Navigation Systems", 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE), pp.331-336, 2023.
- [3] Ze Zhang, Yao Cai, Kristian Ceder, Arvid Enliden, Ossian Eriksson, Soleil Kylander, Rajath Sridhara, Knut Åkesson, "Collision-Free Trajectory Planning of Mobile Robots by Integrating Deep Reinforcement Learning and Model Predictive Control", 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), pp.1-7, 2023.
- [4] Wei Yan, Jian Sun, Zhuo Li, Gang Wang, "Decentralized Multi-Robot Navigation in Unknown Environments via Hierarchical Deep Reinforcement Learning", 2023 42nd Chinese Control Conference (CCC), pp.4292-4297, 2023.
- [5] Silvia Guillén Ruiz, Juan P. Bandera Rubio, Alejandro Hidalgo-Paniagua, Antonio Bandera, "Best Practices for Developing Socially-Aware Service Robots with Predictive Techniques to Enhance Social Comfort", 2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), pp.1-6, 2023.
- [6] Bin Zhang, Junjie Moh, Hun-ok Lim, "Research on Automatic Navigation for a Bipedal Humanoid Robot", 2021 IEEE International Conference on Progress in Informatics and Computing (PIC), 2021, DOI:10.1109/PIC53636.2021.9687086.
- [7] Pengfei Lian, Liang Yuan, Lihui Sun, "Visual navigation for mobile robots based on deep reinforcement learning", 2023 IEEE International Conference on Real-time Computing and Robotics (RCAR), pp.650-656, 2023.
- [8] Ze Zhang, Hadi Hajieghrary, Emmanuel Dean, Knut Åkesson, "Prescient Collision-Free Navigation of Mobile Robots With Iterative Multimodal Motion Prediction of Dynamic Obstacles", IEEE Robotics and Automation Letters, vol.8, no.9, pp. 5488-5495, 2023.
- [9] Haixin Cao, Biao Lu, Huawang Liu, Rui Liu, Xian Guo, "Modeling and MPC-based balance control for a wheeled bipedal robot", 41st Chinese Control Conference, Hefei, China, 2022.
- [10] Nippun Kumaar, Sreeja Kochuvila, "Reinforcement learning based path planning using a topological map for mobile service robot", 2023 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp.1-6, 2023.
- [11] Haiyue Ma, Siqi Wang, Shouwu Zhang, Song Ren, Heng Wang, "Map-less End-to-end Navigation of Mobile Robots via Deep Reinforcement Learning", 2023 IEEE 13th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp.48-52, 2023.
- [12] Mallika Singh, Madan Mohan Rayguru, Gaurav Kaushik, "Reinforcement Learning Based Controllers for Dynamical Systems", 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp.1-4, 2023.
- [13] Bin Zhang, Ryuji Sengoku, Hun-Ok Lim, "Adaptive Motion Control for an Autonomous Mobile Robot Based on Space Risk Map", IEEE Access, vol.11, pp.69553-69562, 2023.
- [14] Yuanwei Bi, Lujian Zhang, Yaowen Liu, Yansen Huang, Hao Liu, "A Local-Global Feature Fusing Method for Point Clouds Semantic Segmentation", IEEE Access, vol.11, pp.68776-68790, 2023.
- [15] Nabih Pico, Beomjoon Lee, Estrella Montero, Meseret Tadese, Eugene Auh, Myeongyun Doh, Hyungpil Moon, "Static and Dynamic Collision Avoidance for Autonomous Robot Navigation in Diverse Scenarios Based on Deep Reinforcement Learning", 2023 20th International Conference on Ubiquitous Robots (UR), pp.281-286, 2023.
- [16] Xiaodong Ma, Weiwei Fang, "Deep Reinforcement Learning Based Multi-Task Automated Channel Pruning for DNNs", 2023 International Joint Conference on Neural Networks (IJCNN), pp.1-9, 2023.
- [17] Stefani Kecman, Dinko Osmankovic, "Perspective Projection Approach to a Faster UAV Navigation and Obstacle Avoidance Learning", 2023 XXIX International Conference on Information, Communication and Automation Technologies (ICAT), pp.1-6, 2023.
- [18] Raffaele Brilli, Marco Legittimo, Francesco Crocetti, Mirko Leomanni, Mario Luca Fravolini, Gabriele Costante, "Monocular Reactive Collision Avoidance for MAV Teleoperation with Deep Reinforcement Learning", 2023 IEEE International Conference on Robotics and Automation (ICRA), pp.12535-12541, 2023.
- [19] Fuchen Kong, Qi Wang, Shang Gao, Hualong Yu, "B-APFDQN: A UAV Path Planning Algorithm Based on Deep Q-Network and Artificial Potential Field", IEEE Access, vol.11, pp.44051-44064, 2023.

- [20] Mauro Martini, Andrea Eirale, Simone Cerrato, Marcello Chiaberge, "PIC4rl-gym: a ROS2 Modular Framework for Robots Autonomous Navigation with Deep Reinforcement Learning", 2023 3rd International Conference on Computer, Control and Robotics (ICCCR), pp.198-202, 2023.
- [21] Tomoaki Nakamura, Masato Kobayashi, Naoki Motoi, "Local Path Planning with Turnabouts for Mobile Robot by Deep Deterministic Policy Gradient", 2023 IEEE International Conference on Mechatronics (ICM), pp.1-6, 2023.
- [22] Julio A. Placed, Jared Strader, Henry Carrillo, Nikolay Atanasov, Vadim Indelman, Luca Carlone, José A. Castellanos, "A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers", IEEE Transactions on Robotics, vol.39, no. 3, pp.1686-1705, 2023.