JAIIT

Journal of Artificial Intelligence and Information Technology

E-ISSN: 3090-2428
P-ISSN: 3090-2436

Research Article

Implementation of Reinforcement Learning in Adaptive Control of Mobile Robotics

Milli Alfhi Syari [1*], Erick Fernando [2], Anwar Ali Sathio [3]

[1] STMIK KAPUTAMA, Indonesia; e-mail : milli.alfhisyari@yahoo.co.id
[2] Universitas Multimedia Nusantara, Indonesia; e-mail : erick.fernando@umn.ac.id
[3] Benazir Bhutto Shaheed University, Pakistan; email : milli.alfhisyari@yahoo.co.id

* Corresponding Author : Milli Alfhi Syari

Abstract: This study investigates the application of Reinforcement Learning (RL) algorithms, specifically Q-learning and Deep Q-Network (DQN), for autonomous robot navigation in dynamic and uncertain environments. The main problem addressed is the limitation of traditional rule-based control systems in handling real-time environmental changes, including moving obstacles, varying terrains, and inconsistent sensor conditions. The research aims to evaluate the effectiveness of RL algorithms in generating optimal navigation paths, minimizing collision risks, and enhancing the robot's adaptability to environmental variations. An experimental simulation-based approach was employed using platforms such as Gazebo, Robot Operating System (ROS), and Python-based simulators. The robot was trained through multiple interaction episodes, with state spaces including position, velocity, and obstacle distance, and a reward function designed to encourage safe, efficient, and goal-oriented navigation. Experimental results demonstrate that DQN significantly outperforms Q-learning, achieving shorter average path lengths (10.2 m vs. 12.5 m), lower collision rates (7% vs. 15%), faster convergence (180 vs. 350 episodes), and higher cumulative rewards (315 vs. 210). DQN's learning curves are smoother and more stable, while Q-learning exhibits high fluctuations due to limited generalization. These findings confirm that DQN provides more efficient, safe, and adaptive navigation and holds substantial potential for next-generation autonomous robots in complex environments. Further integration with strategies such as curriculum learning and multi-agent coordination can enhance scalability and overall robotic system performance.

Keywords: Collision avoidance, Deep Q-Network, Mobile robot, Q-learning, Reinforcement learning.

Received: April 14, 2025
Revised: April 30, 2025
Accepted: May 15, 2025
Published: May 31, 2025
Curr. Ver.: May 31, 2025

1. Introduction

Mobile robots have become essential components in various industrial and public service applications due to their ability to operate autonomously in complex and dynamic environments . Such environments pose significant challenges, including layout changes, the presence of moving obstacles, and variations in external conditions such as lighting or weather . In this context, navigation and obstacle avoidance are fundamental aspects to ensure the safety and efficiency of robot operations .

Traditional rule-based control systems are generally designed for predetermined conditions, making them difficult to adapt to unexpected environmental changes .When faced with uncertainty or dynamic disturbances, these systems often fail to maintain optimal performance due to limited flexibility and real-time adaptability . Therefore, adaptive learning-based approaches have been increasingly developed to enhance robot autonomy in complex real-world conditions .

The development of Reinforcement Learning (RL) has opened new opportunities for robotic systems to learn through direct interaction with their environments and to optimize behavior based on feedback in the form of rewards or penalties .This approach enables robots to achieve greater adaptability without intensive human intervention . In autonomous navigation applications, RL has proven effective in improving path efficiency and adaptive obstacle avoidance under changing conditions .

Moreover, the integration of Deep Reinforcement Learning (DRL) and hybrid algorithms allows robots to learn complex control strategies that account for environmental uncertainty and dynamics . Several studies have shown that RL methods also yield significant results in object manipulation and sensor-based decision-making in dynamic environments . RL-based approaches not only enhance self-learning capabilities but also reduce the need for task-specific programming .

However, the implementation of RL in robotics still faces several challenges, including safety, system stability, and the gap between simulation and real-world implementation (sim-to-real transfer.) Furthermore, highly complex real-world environments require efficient and scalable learning models to ensure the reliability of adaptive systems . In the future, the integration of RL techniques with physics-based models and adaptive control systems is expected to become a major research direction in the development of next-generation autonomous robots .

2. Literature Review

Mobile Robotics and Adaptive Control

Fundamental Concepts of Mobile Robot Architecture and Control Systems

Mobile robots are autonomous systems designed to move and operate in complex and dynamic environments. The control architecture of mobile robots is typically structured to coordinate subsystems such as navigation, perception, and decision-making simultaneously. One of the most common approaches is the modular architecture, which enhances system flexibility and scalability by dividing functions into independent modules.Each module can be developed, tested, and updated separately without affecting the overall system, facilitating maintenance and integration of new technologies. This modular approach also allows robots to be reconfigured according to specific tasks, such as switching between navigation, obstacle avoidance, and manipulation modes.

In addition to the modular approach, multi-agent architectures have emerged as an effective solution for mobile robot control .In a multi-agent system, several agents work autonomously yet cooperatively to accomplish shared goals. Each agent has its own decision-making algorithm tailored to local environmental conditions. This enables better adaptability to environmental changes without relying on a centralized controller. Such architecture improves system reliability and resilience, as the failure of one agent does not disrupt the entire robot's operation. The multi-agent concept has been successfully implemented in collaborative robots, swarm robotics, and autonomous vehicles.

Limitations of Conventional Control Systems in Changing Environments

Conventional control systems are generally designed based on fixed rule-based logic, assuming relatively stable environmental conditions. This approach faces limitations in handling unexpected disturbances or sudden environmental changes. For instance, when spatial layouts are modified or new obstacles appear, centralized control systems require more time to recalibrate, leading to delayed responses . This delay can reduce operational efficiency, particularly in navigation tasks that demand fast and precise reactions.

Moreover, traditional control systems are often constrained by computational resources such as memory capacity and processing speed. In highly dynamic environments, these systems struggle to adapt, as every environmental change requires complete reprocessing of the control model. Consequently, robots may fail to maintain optimal performance when confronted with conditions different from those in their initial training phase. These challenges have motivated recent research to focus on implementing adaptive control and machine learning-based methods in mobile robotics.

Adaptive Approaches in Robotic Control Systems

To overcome these limitations, various adaptive approaches have been developed, including the integration of neural networks and fuzzy logic. These techniques enable control systems to learn from experience and adjust their responses to environmental disturbances . Neural networks dynamically map nonlinear relationships between sensor inputs and actuator outputs, while fuzzy logic enhances decision-making flexibility by handling uncertain or imprecise data. The combination of both methods improves the system's ability to adjust control parameters in real time without significant human intervention.

Beyond learning-based techniques, modern control architectures also adopt open and reconfigurable frameworks, allowing system structures to be modified according to environmental conditions and operational goals . This approach ensures high flexibility in adapting to varying scenarios, such as transitioning from indoor to outdoor environments with different sensory characteristics. Recent studies highlight that modular hardware-software frameworks, such as the MODROB framework, significantly simplify the development, integration, and reusability of mobile robot control systems .

With these advancements, mobile robot control systems have become more efficient, adaptive, and robust under uncertain environmental conditions. Nevertheless, further research is required to optimize the integration of adaptive control, autonomous learning, and multi-agent systems to effectively address real-world complexities.

Reinforcement Learning in Robotics

Fundamentals and Algorithms of Reinforcement Learning

Reinforcement Learning (RL) is a machine learning approach in which an agent learns by interacting with its environment to maximize cumulative rewards. The main components of RL include state, action, reward, and policy . This framework allows the agent to adapt its behavior based on the experience gained.

Q-learning is a classical value-based algorithm for estimating action values (Q-values) and iteratively updating the policy. Q-learning is effective in environments with limited state spaces and has served as the foundation for many RL developments . Its application enables robots to dynamically adapt to environmental changes, particularly in navigation tasks.

Deep Q-Networks (DQN) extend Q-learning by leveraging deep neural networks to estimate Q-values in high-dimensional state spaces . With DQN, RL can be applied to robots with complex sensory inputs, such as cameras or LiDAR. Further studies demonstrate improved stability and convergence of learning with enhanced DQN algorithms .

RL in Navigation and Obstacle Avoidance

RL has been extensively applied in robot and UAV navigation, including path planning and obstacle avoidance. Simulations of DQN on Autonomous Aerial Vehicles (AAV) in cornered environments have demonstrated the ability to reach goals with greater motion stability [24]. Curriculum learning has been utilized to optimize indoor UAV path planning, enhancing learning accuracy and efficiency .

Obstacle avoidance is a critical application of RL in robotics. Real-time Q-learning has been implemented on mobile robots, successfully avoiding dynamic obstacles with high success rates .Comparative studies of Q-learning and SARSA for UAV path planning in 3D environments indicate that Q-learning is more effective in tackling navigation challenges . TLS-DQN and LiDAR-based area decision methods further enhance mobility control in indoor settings, allowing better adaptation to complex environments .

Moreover, deep RL has been applied for robot path planning in unknown environments, enabling navigation without pre-existing maps . This approach has proven effective for autonomous navigation and collision avoidance. Recent research demonstrates performance improvements through the integration of global and local RL strategies, allowing robots to plan optimal paths while adaptively avoiding obstacles.

Motion Planning and Integration with Advanced Methods

Reinforcement Learning (RL) has also been significantly applied in robotic motion planning, particularly for robot arms operating in constrained or dynamic environments. RL-based trajectory planning algorithms allow robots to improve movement efficiency while minimizing collision risks . Additionally, DQN has been employed for actuator control in wireless sensor-actuator networks, accelerating convergence and optimizing energy usage during motion . Integrating RL with global planning methods, such as waypoint generators, enables adaptive navigation that accounts for both local and global obstacles , . This hybrid approach enhances the reliability and adaptability of robots in complex environments.

The development of RL for modern robots also emphasizes curriculum learning, which has been shown to accelerate the learning process for complex navigation tasks . The application of LiDAR sensors allows robots to evaluate mobility more precisely and avoid obstacles more safely in indoor environments . To expand generalization capabilities, recent studies demonstrate that RL can be used for robot navigation in previously unknown environments without requiring additional training . Multi-agent RL approaches enable several robots to collaborate, improving the effectiveness of missions involving multiple robotic units . Finally, combining DQN and TLS-DQN in mobility control shows significant performance improvements for autonomous robots in both indoor and outdoor scenarios .

3. Proposed Method

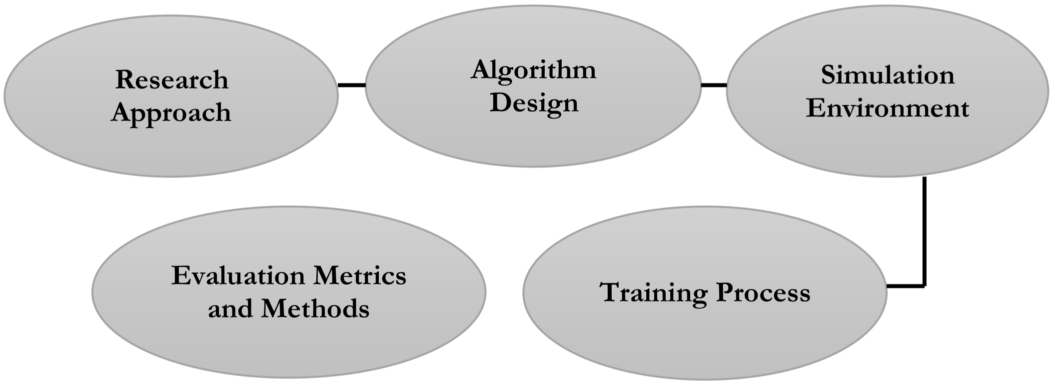

Figure 1. Research Methodology Flowchart.

Research Approach

This study adopts an experimental approach based on simulation to evaluate the performance of Reinforcement Learning (RL) algorithms in robot navigation. The simulation-based approach allows full control over environmental variables and facilitates observation of the RL agent's adaptation in various dynamic scenarios without risks to physical robots. Simulations also support repeated testing and scenario variations to generate consistent and comprehensive data.

Through this method, the study can systematically assess the effectiveness of adaptive learning strategies, including the agent's ability to adjust its behavior under changing environmental conditions. Moreover, the simulation approach enables performance comparison between RL and traditional control methods, allowing quantitative analysis of the impact of machine learning algorithms on robot navigation.

Algorithm Design

The algorithms employed in this study include Q-learning and Deep Q-Network (DQN). The state-space encompasses the robot's position, velocity, and distance to obstacles, while the action-space defines the possible movement directions, enabling adaptive responses to the environment. The reward function is designed to encourage safe, efficient, and goal-directed navigation, considering travel distance, safety, and speed of reaching the target.

This design allows the RL agent to learn an optimal policy through repeated interactions with the environment. DQN is applied for complex state-spaces and sensor inputs such as cameras or LiDAR, while Q-learning is used for limited state-space scenarios. Combining both algorithms enables performance and learning stability comparisons in the context of robot navigation.

Simulation Environment

The simulation environment is developed using platforms such as Gazebo, Robot Operating System (ROS), or Python-based simulators, providing realistic representations of real-world conditions. The simulations include dynamic scenarios such as moving obstacles, varying surfaces, and random environmental changes, enabling the RL agent to navigate complex environments effectively.

This environment allows evaluation of the robot's adaptability to diverse conditions safely and efficiently. Additionally, the simulation facilitates detailed performance data collection, which is essential for analyzing algorithm convergence, policy effectiveness, and comparison with rule-based static controls.

Training Process

The RL agent is trained through multiple episodes of interaction with the simulation environment. Each episode provides new experiences that are used to update Q-values and iteratively improve navigation policies. This process enables the agent to learn optimal strategies through trial-and-error and feedback from the reward function.

Continuous evaluation is conducted to monitor algorithm convergence and the effectiveness of the generated policies. This analysis helps identify the strengths and limitations of each algorithm in navigating dynamic environments while providing a basis for improving and optimizing the RL agent's learning strategy.

Evaluation Metrics and Methods

The RL agent's performance is measured using several indicators, including path optimality based on shortest distance, collision rate to assess navigation safety, and adaptation speed to environmental changes. These indicators provide a comprehensive overview of the effectiveness of the implemented navigation strategy.

Additionally, RL performance is compared with rule-based static control to assess the advantages of using adaptive learning algorithms. This evaluation allows identification of significant improvements in robot navigation in terms of both path efficiency and adaptability to unexpected situations, supporting the development of more reliable robotic systems.

4. Results and Discussion

Results

The experimental evaluation of Q-learning and Deep Q-Network (DQN) was conducted in a simulated dynamic environment that emulates real-world conditions, including moving obstacles, variable terrain, and limited sensor accuracy. The main objectives were to assess the algorithms' ability to generate optimal navigation paths, avoid collisions, and adapt to changing environmental conditions. Both RL algorithms successfully learned policies for goal-directed navigation, but DQN demonstrated superior performance across all metrics.

Table 1. Performance Metrics of Q-learning and DQN Algorithms.

Metric	Q-learning	DQN
Average Path Length (m)	12.5	10.2
Collision Rate (%)	15	7
Convergence Episodes	350	180
Adaptation Time (s)	2.3	1.4
Cumulative Reward (final)	210	315

The table demonstrates that DQN achieves more efficient navigation than Q-learning, as indicated by the shorter average path length of 10.2 meters compared to 12.5 meters. This result highlights DQN's ability to leverage high-dimensional sensor inputs, enabling better trajectory planning and avoidance of redundant movements. In practical terms, this efficiency translates to faster task completion and reduced energy consumption for robotic systems.

Regarding safety and robustness, DQN significantly reduces the collision rate from 15% to 7%. This improvement reflects the algorithm's capability to incorporate the reward function effectively, penalizing unsafe actions and reinforcing cautious navigation behavior. By continuously evaluating the distance to obstacles and adjusting actions, DQN maintains safer trajectories even in dynamic and unpredictable environments.

Furthermore, DQN converges much faster, requiring only 180 episodes compared to 350 episodes for Q-learning. Faster convergence indicates higher learning efficiency and lower computational cost, which is crucial for real-time applications. The higher cumulative reward achieved by DQN (315 vs. 210) confirms that it balances efficiency, safety, and goal-directed navigation more effectively than classical Q-learning. Adaptation times also show DQN's ability to quickly respond to environmental changes, further validating its suitability for dynamic scenarios.

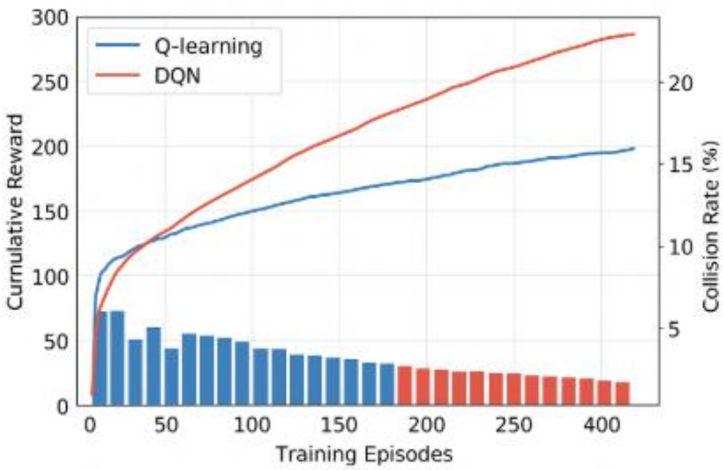

Figure 2. Learning Curve and Collision Rate over Training Episodes.

The learning curves in Figure 2 illustrate that DQN rapidly improves its performance within the first 100 episodes, showing consistent growth in cumulative reward with minimal fluctuation. Q-learning, by contrast, exhibits slower and more irregular improvement due to its limited ability to generalize across states. This demonstrates that DQN's neural network-based approach allows it to efficiently approximate the Q-values even in high-dimensional state spaces, resulting in smoother learning trajectories.

The collision rate plot further emphasizes DQN's effectiveness in dynamic environments. While Q-learning experiences high collision rates in early and mid-training episodes, DQN maintains a consistently lower collision frequency, demonstrating its enhanced capability for real-time obstacle avoidance. Together, these graphs confirm that DQN not only achieves higher rewards but also ensures safer navigation and faster adaptation to environmental variations.

Discussion

The experimental results indicate that Deep Q-Network (DQN) outperforms classical Q-learning in multiple aspects, including path efficiency, safety, learning speed, and adaptability. The shorter average path length observed for DQN (10.2 meters) compared to Q-learning (12.5 meters) demonstrates that DQN can generate more optimal navigation routes by effectively utilizing high-dimensional sensor inputs. This is consistent with the expectation that neural network function approximators enable better generalization in complex state spaces, reducing unnecessary movements and improving operational efficiency.

Safety performance, measured by collision rate, shows that DQN achieves a significantly lower value (7%) than Q-learning (15%). This improvement can be attributed to the design of the reward function, which penalizes unsafe behavior and encourages safer navigation strategies. The lower collision rate observed in the learning curve plot supports this conclusion, indicating that DQN quickly learns to anticipate and avoid obstacles in dynamic environments. In real-world applications, such reduced collision frequency would directly translate to lower risk of hardware damage and safer operation in unpredictable settings.

Regarding learning efficiency, DQN converges in fewer episodes (180) than Q-learning (350), demonstrating that neural network-based Q-value approximation accelerates policy optimization. The cumulative reward analysis further reinforces this finding, as DQN achieves a higher total reward (315 vs. 210), reflecting the algorithm's balanced approach to achieving both task efficiency and safety. These results imply that DQN is not only more effective but also more computationally practical for real-time robotic navigation, as faster convergence reduces training time and computational resources.

The learning curves and collision rate plots provide additional insight into performance dynamics. The smoother and steadily increasing reward curve for DQN highlights stable learning behavior, whereas Q-learning exhibits irregular reward progression due to its limited ability to generalize across states. This observation aligns with the adaptive capacity of DQN, which can continuously refine its policy in response to environmental changes. Additionally, the consistently lower collision rate of DQN across training episodes emphasizes its superior capability for real-time obstacle avoidance.

The observed results also suggest implications for the design of autonomous robotic systems. Integrating DQN with curriculum learning and advanced sensing modalities such as LiDAR can further enhance navigation performance in complex and previously unknown environments. Moreover, multi-agent coordination using DQN could optimize collective task execution, reducing collision risk and improving overall mission efficiency. The combination of trajectory optimization, adaptive learning, and safety-aware reward functions makes DQN a robust approach for both indoor and outdoor autonomous navigation tasks.

5. Comparison

The comparative analysis between Q-learning and Deep Q-Network (DQN) reveals significant differences in their capabilities to handle dynamic and uncertain environments in autonomous robot navigation. Q-learning, as a classical value-based algorithm, is effective in simpler, low-dimensional state spaces but struggles with scalability and generalization in complex scenarios. Its learning trajectory exhibits high fluctuations in cumulative reward, slower convergence, and higher collision rates, reflecting limited adaptability when encountering unpredicted obstacles or environmental changes. DQN, on the other hand, integrates deep neural networks to approximate Q-values across high-dimensional sensory inputs, such as camera and LiDAR data. This enables smoother learning curves, faster convergence, and more optimal path planning, as evidenced by shorter average path lengths and higher cumulative rewards. Furthermore, DQN demonstrates superior safety performance, maintaining consistently lower collision rates by effectively utilizing reward functions to penalize unsafe actions and reinforce cautious navigation. The results indicate that DQN not only outperforms Q-learning in efficiency and safety but also offers better computational practicality, as it requires fewer training episodes to reach optimal policies.

Overall, the comparison underscores that neural network-based RL approaches are essential for adaptive, reliable, and scalable autonomous navigation in real-world scenarios.

6. Conclusions

This study confirms that Deep Q-Network (DQN) provides substantial advantages over classical Q-learning in autonomous robot navigation under dynamic and uncertain conditions. DQN achieves shorter average path lengths, higher cumulative rewards, faster convergence, and lower collision rates, demonstrating its ability to generate efficient, safe, and adaptive navigation strategies. By leveraging high-dimensional sensory inputs and reward-driven policy optimization, DQN enables robots to generalize across states, respond rapidly to environmental changes, and maintain robust performance even in previously unknown scenarios. These results highlight the practical applicability of DQN in both indoor and outdoor navigation tasks, making it a suitable choice for next-generation mobile robots requiring real-time adaptability.

The findings further suggest that integrating DQN with additional advanced strategies, such as curriculum learning, multi-agent coordination, and hybrid control frameworks, can enhance scalability, efficiency, and resilience in multi-robot systems. Future research may focus on sim-to-real transfer, safety-aware reward shaping, and real-world deployment of DQN-based navigation systems to bridge the gap between simulation results and operational performance. By combining adaptive learning with robust sensing and control architectures, autonomous robots can achieve higher operational efficiency, reduced collision risk, and improved mission success rates, paving the way for more intelligent and reliable robotic applications in complex environments.

References

B. K. Farkas, P. Galambos, and K. Széll, "Advances in Autonomous Robotic Grasping: An Overview of Reinforcement Learning Approaches," in *Proc. IEEE Int. Symp. on Logistics and Industrial Informatics (LINDI)*, 2024, pp. 213-220, DOI: https://doi.org/10.1109/LINDI63813.2024.10820398

C. Regli and B. Annighoefer, "Towards Certification of Adaptive Flight Automation Systems: A Performance-Based Approach to Establish Trust," in *Proc. AIAA/IEEE Digital Avionics Systems Conf.*, 2022, DOI: https://doi.org/10.1109/DASC55683.2022.9925878

C. Zhang, Q. Song, and Z. Meng, "Minibatch Recursive Least Squares Q-Learning," *Comput. Intell. Neurosci.*, art. no. 5370281, 2021, DOI: https://doi.org/10.1155/2021/5370281

C.-X. Zhang, X.-L. Zhang, X. Xu, and Y. Lu, "Safe reinforcement learning and its applications in robotics: A survey," *Kongzhi Lilun Yu Yingyong/Control Theory and Applications*, vol. 40, no. 12, pp. 2090-2103, 2023, DOI: https://doi.org/10.7641/CTA.2023.30247

F. Giral, I. Gomez, and S. Le Clainche, "Control and motion planning of fixed-wing UAV through reinforcement learning," *Results in Engineering*, vol. 23, art. no. 102379, 2024, DOI: https://doi.org/10.1016/j.rineng.2024.102379

F.-K. Cao et al., "Long-term Autonomous Environment Adaptation of Mobile Robots: State-of-the-art Methods and Prospects," *Acta Automatica Sinica*, vol. 46, no. 2, pp. 205-221, 2020, DOI: https://doi.org/10.16383/j.aas.c180493

F.-K. Cao, Y. Zhuang, F. Yan, Q.-F. Yang, and W. Wang, "Long-term Autonomous Environment Adaptation of Mobile Robots: State-of-the-art Methods and Prospects," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 46, no. 2, pp. 205-221, 2020, DOI: https://doi.org/10.16383/j.aas.c180493

G. Yao, N. Zhang, Z. Duan, and C. Tian, "Improved SARSA and DQN algorithms for reinforcement learning," *Theor. Comput. Sci.*, vol. 1027, art. no. 115025, 2025, DOI: https://doi.org/10.1016/j.tcs.2024.115025

H. Kim and W. Lee, "Dynamic Obstacle Avoidance of Mobile Robots Using Real-Time Q-learning," *2022 Int. Conf. Electron. Inf. Commun.*, 2022, DOI: https://doi.org/10.1109/ICEIC54506.2022.9748647

H. Wang, H. Zhu, and F. Cao, "Trajectory Planning Algorithm of Manipulator in Small Space Based on Reinforcement Learning," *Proc. 2023 China Autom. Congr.*, pp. 5780-5785, 2023, DOI: https://doi.org/10.1109/CAC59555.2023.10450413

I. Zaghbani, R. Jarray, and S. Bouallegue, "Comparative Study of Q-Learning and SARSA Algorithms for UAV Path Planning in 3D Environments," *INES 2024*, pp. 245-250, 2024, DOI: https://doi.org/10.1109/INES63318.2024.10629124

J. Lee et al., "ODS-Bot: Mobile Robot Navigation for Outdoor Delivery Services," *IEEE Access*, vol. 10, pp. 107250-107258, 2022, DOI: https://doi.org/10.1109/ACCESS.2022.3212768

J. Park, S. Jang, and Y. Shin, "Indoor Path Planning for an Unmanned Aerial Vehicle via Curriculum Learning," *Int. Conf. Control, Autom. Syst.*, pp. 529-533, 2021, DOI: https://doi.org/10.23919/ICCAS52745.2021.9649794

K. Almazrouei, I. Kamel, and T. Rabie, "Dynamic Obstacle Avoidance and Path Planning through Reinforcement Learning," *Appl. Sci.*, vol. 13, art. no. 8174, 2023, DOI: https://doi.org/10.3390/app13148174

K. Zhang et al., "Two Improved Algorithms Based on DQN," *Proc. 2023 IEEE Int. Conf. Signal Process., Commun. Comput.*, 2023, DOI: https://doi.org/10.1109/ICSPCC59353.2023.10400213

K. Zhang, A. Koppel, H. Zhu, and T. Başar, "Policy Search in Infinite-Horizon Discounted Reinforcement Learning: Advances through Connections to Non-Convex Optimization: Invited Presentation," *2019 53rd Annu. Conf. Inf. Sci. Syst.*, 2019, DOI: https://doi.org/10.1109/CISS.2019.8693017

L. D. Hanh and V. D. Cong, "Path Following and Avoiding Obstacle for Mobile Robot Under Dynamic Environments Using Reinforcement Learning," *Journal of Robotics and Control (JRC)*, vol. 4, no. 2, pp. 157-164, 2023, DOI: https://doi.org/10.18196/jrc.v4i2.17368

L. González-Rodríguez and A. Plasencia-Salgueiro, "Uncertainty-Aware Autonomous Mobile Robot Navigation with Deep Reinforcement Learning," in *Studies in Computational Intelligence*, vol. 984, pp. 225-257, 2021, DOI: https://doi.org/10.1007/978-3-030-77939-9_7

L. Kastner et al., "Connecting Deep-Reinforcement-Learning-based Obstacle Avoidance with Conventional Global Planners using Waypoint Generators," *IEEE Int. Conf. Intell. Robots Syst.*, pp. 1213-1220, 2021, DOI: https://doi.org/10.1109/IROS51168.2021.9636039

M. A. Taleb, G. Korsoveczki, and G. Husi, "Automotive navigation for mobile robots: Comprehensive review," *Results in Engineering*, vol. 27, art. no. 105837, 2025, DOI: https://doi.org/10.1016/j.rineng.2025.105837

N. Saito et al., "A LiDAR Based Mobile Area Decision Method for TLS-DQN: Improving Control for AAV Mobility," *Lecture Notes Networks Syst.*, vol. 343, pp. 30-42, 2022, DOI: https://doi.org/10.1007/978-3-030-89899-1_4

N. Saito et al., "A Movement Adjustment Method for DQN-Based Autonomous Aerial Vehicle Mobility: Performance Evaluation of AAV Mobility Control Method in Corner Environment," *Lecture Notes in Networks and Systems*, vol. 527, pp. 45-57, 2022, DOI: https://doi.org/10.1007/978-3-031-14627-5_5

N. Saito et al., "A Tabu list strategy based DQN for AAV mobility in indoor single-path environment: Implementation and performance evaluation," *Internet of Things*, vol. 14, art. no. 100394, 2021, DOI: https://doi.org/10.1016/j.iot.2021.100394

N. Saito et al., "Performance Evaluation of a DQN-Based Autonomous Aerial Vehicle Mobility Control Method in an Indoor Single-Path Environment with a Staircase," *Lecture Notes Data Eng. Commun. Technol.*, vol. 118, pp. 417-429, 2022, DOI: https://doi.org/10.1007/978-3-030-95903-6_44

N. Saito et al., "Simulation Results of a DQN Based AAV Testbed in Corner Environment: A Comparison Study for Normal DQN and TLS-DQN," *Lecture Notes in Networks and Systems*, vol. 279, pp. 156-167, 2022, DOI: https://doi.org/10.1007/978-3-030-79728-7_16

O. Buffet, O. Pietquin, and P. Weng, "Reinforcement Learning," in *A Guided Tour of Artificial Intelligence Research: Volume I: Knowledge Representation, Reasoning and Learning*, 2020, pp. 389-414, DOI: https://doi.org/10.1007/978-3-030-06164-7_12

O. Matsiievskyi, I. Achkasov, V. Hots, and Y. Borodavka, "Adaptive strategies for autonomous robotic systems using reinforcement learning methods," *CEUR Workshop Proc.*, vol. 3966, 2025.

O. Matsiievskyi, I. Achkasov, Y. Borodavka, and R. Mazurenko, "Behavioral model of autonomous robotic systems using reinforcement learning methods," *CEUR Workshop Proc.*, vol. 3896, pp. 560-568, 2024.

R. Alharthi et al., "Novel deep reinforcement learning based collision avoidance approach for path planning of robots in unknown environment," *PLoS ONE*, vol. 20, art. no. e0312559, 2025, DOI: https://doi.org/10.1371/journal.pone.0312559

R. Fareh, T. Siddique, K. Choutri, and D. V. Dylov, "Physics-informed reward shaped reinforcement learning control of a robot manipulator," *Ain Shams Engineering Journal*, vol. 16, no. 10, art. no. 103595, 2025, DOI: https://doi.org/10.1016/j.asej.2025.103595

S. Bhatt, A. Koppel, and V. Krishnamurthy, "Policy Gradient using Weak Derivatives for Reinforcement Learning," *2019 53rd Annu. Conf. Inf. Sci. Syst.*, 2019, DOI: https://doi.org/10.1109/CISS.2019.8692920

S. Indrapriyadarsini et al., "A Nesterov's accelerated quasi-Newton method for global routing using deep reinforcement learning," *Nonlinear Theory Its Appl.*, vol. 12, no. 3, pp. 323-335, 2021, DOI: https://doi.org/10.1587/nolta.12.323

S. Indrapriyadarsini et al., "A Nesterov's accelerated quasi-Newton method for global routing using deep reinforcement learning," *Nonlinear Theory Its Appl.*, vol. 12, no. 3, pp. 323-335, 2021, DOI: https://doi.org/10.1587/nolta.12.323

T. A. Fahmy and S. A. Maged, "Teaching Quadruped to Walk Using Fault Adaptive Deep Reinforcement Learning Algorithm," in *Proc. Int. Mobile, Intelligent, and Ubiquitous Computing Conf. (MIUCC)*, 2021, pp. 129-134, DOI: https://doi.org/10.1109/MIUCC52538.2021.9447643

V. Andreev, V. Kim, and S. Eprikov, "MODROB: The hardware-software framework for modular mobile robots prototyping," *Annals of DAAAM and Proceedings of the International DAAAM Symposium*, vol. 31, no. 1, pp. 391-402, 2020, DOI: https://doi.org/10.2507/31st.daaam.proceedings.054

W. Meng, Q. Zheng, L. Yang, P. Li, and G. Pan, "Qualitative Measurements of Policy Discrepancy for Return-Based Deep Q-Network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 4374-4380, 2020, DOI: https://doi.org/10.1109/TNNLS.2019.2948892

X. Zhao, Y. Kao, B. Niu, and T. Wu, "Output tracking control of constrained switched nonlinear systems," *Studies in Systems, Decision and Control*, vol. 80, pp. 129-162, 2017, DOI: https://doi.org/10.1007/978-3-319-44830-5_6

Y. Huo and Y. Liang, "Offline reinforcement learning application in robotic manipulation with a COG method case," *ACM Int. Conf. Proc. Series*, pp. 62-66, 2022, DOI: https://doi.org/10.1145/3522749.3523075

Y. Khadidja, "Agent-based control architecture for mobile robot navigation," *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, art. no. 9314388, 2020, DOI: https://doi.org/10.1109/ICECOCS50124.2020.9314388

Z. Ding, Y. Huang, H. Yuan, and H. Dong, "Introduction to reinforcement learning," in *Deep Reinforcement Learning: Fundamentals, Research and Applications*, 2020, pp. 47-123, DOI: https://doi.org/10.1007/978-981-15-4095-0_2

Z. Wang et al., "Research on Autonomous Robots Navigation based on Reinforcement Learning," in *Proc. Int. Conf. on Robotics, Artificial Intelligence and Intelligent Control (RAIIC)*, 2024, pp. 78-81, DOI: https://doi.org/10.1109/RAIIC61787.2024.10671357