Research on a Dynamic Path Planning Algorithm for Robots Based on Deep Reinforcement Learning

Boyi Zheng

School of Physics, Harbin Institute of Technology, Harbin, China

18312974485@163.com

Abstract. With the advancement of robotic technology, robot path planning has garnered significant attention, particularly in fields such as automation, autonomous driving, and smart manufacturing. Path planning involves determining the optimal path for a robot to travel from a specific position to its destination while avoiding obstacles. Traditional path planning algorithms can effectively realize robot path planning in a static environment where obstacles do not move. In practice, some environments can change rapidly, and unexpected obstacles may arise. Traditional path planning algorithms often struggle to optimize paths in dynamic and complex real-world applications adaptively. Hence, the performance of traditional path planning algorithms is limited. This paper presents a robot path planning algorithm in a complex dynamic environment using deep reinforcement learning (DRL). Using Proximal Policy Optimization (PPO), reinforcement learning enables the robot to learn path planning methods by simulating its interactions with its environment. As the robot runs autonomously, continuously improving its experience, our proposed dynamic path planning can adaptively optimize paths according to the movement of obstacles. The experimental results confirm that the proposed DRL-based approach enables the robot to reach the destination faster by avoiding moving obstacles and achieving a better performance than conventional path planning approaches when the surrounding environment fluctuates unexpectedly. Therefore, the experiments suggest that the proposed algorithm enhances robots' ability to adapt to complex dynamic environments across a broad range of applications, including, but not limited to, autonomous vehicles, mobile robots, and automated industrial systems.

Keywords: Deep Reinforcement Learning, Path Planning, Dynamic Environment, Obstacle Avoidance, Robotics.

1. Introduction

Robotic systems are becoming increasingly popular in various fields, including automation, logistics, manufacturing, healthcare, and the service sector. As these robot systems become more sophisticated, they often need to interact with complex and unpredictable environments, which involve path planning as a key function. Path planning is finding an optimal route from the starting point to the destination while avoiding obstacles and navigating around rugged terrain. Traditional path planning algorithms, such as A* and D*, are highly effective in static environments where obstacles are known and the environment remains unchanged. These methods can create maps of their surrounding environment, determine the shortest or most efficient path, and then navigate based on those predetermined paths. However, in the world we live in, environments are rarely static. Dynamic obstacles, such as pedestrians, vehicles, or heavy machinery, that move like pedestrians can add complexity to path planning problems. In such dynamic environments, the traditional approach fails because the robot must continuously update its route when a new obstacle appears [7][8]. Deep Reinforcement Learning (DRL) provides a promising alternative to overcome the limitations of traditional path planning algorithms. Unlike the pre-programmed approach, which instructs robots on the best way to find the shortest route, DRL enables robot paths to be learned through trial and error within their environment. In the context of robot navigation, robots can adjust their path planning strategy constantly in response to moving obstacles and other environmental conditions, learning over time how to plan safer routes in dynamic environments [2]. This study presents a dynamic path planning algorithm for robots utilizing reinforcement learning, specifically the proximal policy optimization (PPO) algorithm. The PPO algorithm is one of the latest and most efficient algorithms among all deep reinforcement learning techniques, and it has several advantages, including fast convergence and stability of optimization processes in various complex and high-dimensional environments. The algorithm proposed in this paper enables robots to adjust their paths in real-time, ensuring safe movement in complex environments with dynamic obstacles.

2. Literature Review

2.1 Traditional and Reactive Path Planning Methods

The problem of robot path planning has been a focus of research in robotics. The study was once focused on solving the path planning problem in a static environment. Traditional path planning algorithms include A* algorithm and Dijkstra's algorithm. Both algorithms assume the environment is static, meaning the obstacles in the path do not change after the route is calculated. These path planning algorithms are suitable for the well-known environment and structured scenes. To compensate for these shortcomings, reactive methods such as Potential Field Methods and the Bug Algorithm were introduced directly into real environments without the need for pre-computed maps. They could react quickly and continuously change the robot's trajectory depending on information from the sensors. However, they can sometimes become trapped in local minima (the robot enters a bad state) or be unable to move around obstacles, resulting in an oscillating path [3].

2.2 The Rise of Machine Learning and Deep Reinforcement Learning

With the introduction of machine learning, particularly Reinforcement Learning (RL), researchers have begun exploring new methods for robots to learn effective path planning strategies through observation, rather than traditional logic-based reasoning and search methods. In RL, the agent learns navigation by interacting with the environment using reward/penalty feedback through the action space. RL combined with Deep Learning (DL), known as Deep Reinforcement Learning (DRL), has been recently applied to robot path planning, as DRL is well-suited for complex environments that most classical path planning algorithms cannot handle. One of the well-researched DRL techniques is called Deep Q-Learning (DQN). It is a reinforcement learning technique that can successfully simulate human experience and learn how to take the best actions through trial and error in an environment. It is widely used in video games, robotics, and vehicle control. The core idea is based on Deep Q-Learning (DQN) and its application to robotic control, which requires a neural network training to find an optimal solution that the Q function (the value function of expected future reward of an action taken at that time) can estimate, which means the action of the robot can be calculated by an equation if in a particular state. However, DQN performs less effectively on tasks involving continuous actions.

2.3 Advancements in Policy Optimization and Hybrid Algorithms

Furthermore, it incurs higher computing costs. A relatively new DRL technique is Proximal Policy Optimization(PPO). PPO aims to provide a simple, flexible, and general implementation of deep stochastic policy gradient methods that improve upon the performance of DQN with respect to speed and efficiency when agents are training and learning policies. It offers a wide variety of potential applications, particularly for robotics [2]. PPO can achieve high stability and performance across several tasks, compared to previous state-of-the-art methods. More importantly, the optimization process is less sensitive to hyperparameters and allows for better convergence.

2.4 Multi-Sensor Fusion for Enhanced Perception in DRL

In addition to the above approaches based on DRL for path planning, researchers have also explored multi-sensor inputs in combination with DRL, because integrating signals from various types of sensors can help robots obtain a comprehensive view of the surrounding environment, allowing path planning for complicated or dynamic scenarios [3], such as lidar. By utilizing heterogeneous signals simultaneously from different sensor types, this multimodal perception can significantly reduce uncertainty, thereby increasing the likelihood that a mobile robot avoids collisions with obstacles while navigating complex terrain. In summary, this thesis uses the DRL approach for the robot's dynamic obstacle avoidance. It combines several heterogeneous signal sources via embedded hardware circuits for sensor perception on a mobile platform.

3. Algorithm Principle

Other researchers have developed a dynamic path planning algorithm for robots using Deep Reinforcement Learning (DRL). It utilizes the Proximal Policy Optimization (PPO) algorithm to train the robot to navigate a complex and dynamic environment in real-time [2]. The core idea of this method is to utilize DRL and combine it with multi-modal perception information to enhance the sensor's capabilities, enabling the robot to adapt to environmental changes while continuously refining its movement route. This section describes the key components of this idea, which mainly include: environment establishment, DRL training, multi-sensory fusion, and performance evaluation.

3.1 Environment Modeling

The environment is simulated in a grid-based model with several static and dynamic moving obstacles. Static obstacles could be as simple as walls or as complex as a long bridge spanning a deep valley or river. Dynamic obstacles allow the robot to simulate real-world unpredictability, ranging from small vehicles like cars or bikes to more complex robots, or even people walking in and out of different paths of an already complex map. These objects change their position over time and are moved throughout the simulation at the same speed as their intended purpose would dictate. Therefore, these obstacles must be updated at every step of the simulation to reflect their changing positions.

3.2 Deep Reinforcement Learning Training

The key to the path planning algorithm is the Deep Reinforcement Learning (DRL) algorithm, in which the robot learns to take actions based on feedback from the environment. In the DRL environment, the interaction between agents and environments is considered as follows. The objective is that during training, the robot can take optimal actions over a long series in response to the given environment based on its sensory data, making it achieve the objective (which refers to moving from the start to the destination without obstacles), which can be achieved by getting as many maximum cumulative rewards as possible. The algorithm utilizes the Proximal Policy Optimization (PPO) algorithm for training the robot policy, which falls under the category of continuous action space (robot navigation) reinforcement learning models. The PPO employs clipped surrogate optimization, an on-policy reinforcement learning algorithm that utilizes a surrogate objective function to facilitate gradient descent and promote stable learning. The use of such surrogate functions helps optimize neural networks safely and effectively. PPO has several advantages over other algorithms in DRL models, including better sample efficiency and greater stability in environments with large state spaces. During the DRL training process, the robot's current observation determines the action it takes. Each observation will be treated as a current state (the coordinate and velocity of the robot’s position, the distances to obstacles ahead or around). At each time step, the robot takes an action based on its observation state (such as moving forward, left, or right). According to the environment status (the surroundings and obstacles) resulting from its current situation and its choice of actions, the robot receives a reward (or penalty) in different cases. A positive reward is given to the robot when it moves closer to the target location without colliding with an obstacle. A negative reward is given to the robot if it collides or deviates from the optimal route. PPO utilizes a clipped objective function. Its clip limits the update size in the training period, so there would be large jumps or chaotic changes during policy optimization. The trained policy is improved iteration after iteration using batches (or mini-batches) of experience acquired through agent-environment interactions. Through repeated actions during training, the agent learns to refine its optimal path planning strategy through exploration and refinement of samples, enabling efficient obstacle avoidance and target reaching.

3.3 Multi-modal Perception Fusion

The author designs several different types of sensors in the research environment to enhance the robot's perception and decision-making ability [4]. Multi-modal sensors provide information on obstacles and robots, which is more comprehensive than that of any single sensor mode. This information can be fused with sensor fusion technology to provide a more objective and reasonable way to describe the surroundings. In the study, researchers designed a robot equipped with LiDAR (Light Detection and Ranging), Cameras, and Radars. The LiDAR provides accurate distance measurements to detect obstacles in the robot’s vicinity, allowing the robot to obtain a panoramic view of its environment. The cameras can detect visual information of objects in front of the vehicle to classify them, such as detecting a dynamic obstacle (e.g., a moving car or pedestrian), which helps the robot understand the scene around it and anticipate potential collisions with obstacles. The radars capture images in different ways to generate environmental images with varying details for the robot, utilizing radar technology.

Additionally, the sensors, including the acceleration sensor and GPS, can also be configured. And then, it extracts features with a deep neural network algorithm DNN[4]. It uses the PPO-based DRL algorithm to learn from experience data. The data from these several sensors, after extracting feature vectors by the DNN algorithm, are input into the PPO-based DRL algorithm. Finally, the reward of driving through this environment can be obtained.

3.4 Evaluation Methodology

We can simulate the performance of the proposed DRL-based algorithm in a scenario where the robot must move from the starting point to the goal while avoiding obstacles along the way. Evaluation of how well the robot avoids obstacles, especially when dealing with a moving obstacle, is necessary. The path should be adjusted to accommodate objects that appear or disappear. We also assess how the robot completes its task to optimize its path while avoiding obstacles, ensuring it functions efficiently in practical, real-world applications. Furthermore, the robot must adapt to changing situations, such as the appearance of new obstacles, the movement of existing obstacles, or the detection of new ones. In other words, the robot needs to adapt quickly and effectively to change direction and avoid newly discovered obstacles to reach its target destination. To test these behaviors, simulations are implemented through comparative experiments where our DRL-based algorithm is applied alongside several well-known algorithms, including A* and D*, under dynamic conditions where obstacles are varied and sometimes change. A*, D* work quite well in static environments, while the re-planning may involve finding another shortest path from the exact origin if any obstacle appears or disappears.

In contrast, we designed a model that enables robots to avoid obstacles while dynamically searching for the shortest path by learning from experience, eliminating the need to plan the entire route. Besides, various conditions are involved. For example, the densities of obstacles are different, as are their dynamical behaviors and changes in moving obstacles. It is expected that DRL can learn to navigate through obstacle-free areas more efficiently, minimize the distance covered, and respond to dynamic changes to reach the final position.

4. DRL Performance

Finally, the performance of the DRL algorithm used in this paper surpasses that of traditional algorithms in obstacle avoidance. When moving around obstacles, the robot using the DRL algorithm can adjust its moving direction in real time to avoid collisions with dynamic obstacles and quickly arrive at target points by taking paths that minimize the distance. The traditional algorithm will generate a static path before motion. When an object that moves at an unpredictable speed or an environment-changing object appears before the robot, it can only return once or collide with the object.

4.1 Obstacle Avoidance Performance

The DRL-based algorithm demonstrated superior obstacle avoidance capabilities compared to traditional methods. In dynamic environments, where obstacles moved unpredictably, the robot using the DRL approach was able to adjust its path in real-time, avoiding collisions and reaching its target efficiently. Traditional algorithms, which rely on pre-computed paths, struggle to adapt to environmental changes and often result in collisions or suboptimal paths.

4.2 Path Optimization

The DRL-based algorithm also outperforms traditional ones in path optimization. A robot can find a shorter and smoother path, especially compared to those of A* and D*. This would be especially beneficial in cases where there are numerous dynamic obstacles, requiring the robot to adjust its motion route dynamically, which in turn reduces the total travel distance and leads to improved performance.

4.3 Real-time Adaptation

Furthermore, similar to obstacle avoidance and path optimization, another advantage of the DRL algorithm is its ability to learn new obstacles in real-time. Suppose new obstacles or changes are encountered while the robot is in motion. Therefore, the DRL algorithm could adjust the path taken by the robot in real-time to avoid the barriers. This function enables DRL algorithms to adapt to changes in surrounding obstacles continually and recalculates paths in response to changes in obstacle status, allowing the robot to avoid collisions while continuously exploring. Unlike traditional planning algorithms, which can only detect the emergence of new obstacles on routes when computing paths, DRL algorithms can be continuously learned in this aspect [8].

5. Discussion

Although the path planning algorithm based on DRL yielded good results, several challenges and problems remain that require further research and testing. One challenge is the cost of training the agent of DRL. Teaching a robot to navigate through an environment with complex obstacles requires numerous iterations, especially in large-scale environments that contain more dynamic objects in the future. The next step can focus on optimizing and reducing the computational requirements to make the DRL approach more computationally efficient and scalable. Moreover, although integrating sensors of different modalities improves the accuracy of robot localization and mapping, the quality of the sensor data is problematic. In real-time applications, a robot must handle complex data and signals that can fluctuate and be influenced by factors in real-world environments. For instance, lidar sometimes fails to detect objects due to weather or dust, while cameras suffer from issues such as lighting conditions. Different methods for sensor fusion can be tested to extract a clearer image of the world around the robot and address the problems caused by sensor limitations, inaccuracies, and imperfections [3]. Although PPO provides a solid method based on DRL to solve the issue of dynamic path planning for UGVs, it isn't the only option available for testing in future research laboratories. There may be more efficient approaches that DRL researchers have proposed, such as Deep Deterministic Policy Gradient (DDPG) and Trust Region Policy Optimization (TRPO), which may yield better performance results [2].

6. Conclusion

This paper proposes a robot dynamic path planning algorithm based on DRL, using the Proximal Policy Optimization (PPO) algorithm, which enables the robot to plan paths in a complex environment, can avoid obstacles in real time, and find a better path. Our proposed DRL-based dynamic path planning algorithms surpass traditional methods, such as the A* algorithm or D* algorithm, in various situations. Even in more complex moving obstacle environments, robots can plan a better dynamic path. We also present our experimental results to demonstrate the advantages of using DRL for the robot's motion path planning compared to existing articles. Our experiment indicates that our DRL methods enable robots to continuously adapt to environmental changes and avoid collisions without recomputing a new path for obstacle avoidance, unlike some classic approaches that typically require this. This is particularly important, especially in the context of real-time path planning. Furthermore, our multi-sensor fusion can utilize one or more sensor input sources, such as LiDAR and a camera. It will enable the robot to have a better perception and decision-making capability in obstacle detection. Although we achieve excellent performance with the DRL method, there is still more work to be done. For example, training costs remain high, and it is challenging to train in a large-scale environment. Therefore, it is necessary to reduce computational efforts to train and update algorithms for practical applications. Besides sensor limitations, they may pose some challenging problems for detecting various obstacles when robots are deployed in long-term, complex, and unknown environments. It needs more sensors or the development of other sensor algorithms to optimize. In the future, our work will focus on developing and improving sensor fusion methods. Reducing the time required for learning DRL strategies by integrating other optimized techniques, and seeking alternative algorithms to further enhance our DRL robotic motions in dynamic environments, etc.

References

[1] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver & Daan Wierstra. Continuous control with deep reinforcement learning. 2016.

[2] John Schulman, et al. Proximal Policy Optimization Algorithms. 2017.

[3] Shulan Ruan, Rongwei Wang, Xuchen Shen, Huijie Liu, Baihui Xiao, Jun Shi, Kun Zhang, Zhenya Huang, Yu Liu, Enhong Chen, You He. A Survey of Multi-sensor Fusion Perception for Embodied AI: Background, Methods, Challenges and Prospects. 2025.

[4] Saeid Nahavandi, et al. A Comprehensive Review on Autonomous Navigation. 2025.

[5] Jingyuan Zhao, et al. A Survey of Autonomous Driving from a Deep Learning Perspective. 2025.

[6] Volodymyr Mnih, et al. A Survey of Autonomous Driving from a Deep Learning Perspective. 2015.

[7] YANG Longfei, LAI Huicheng, DU Haohao, ZHANG Guo. Research on Optimization Algorithms for Multi-Robot Path Planning. 2025.

[8] Kun Zhang, Yongfu Shi. Adaptive Path Planning Method for Robots in Complex Environments. 2025.