Goal-Driven Autonomous Exploration Through Deep Reinforcement Learning

Kanishk Kaushal 002306180

Abstract

This project introduces a goal-driven autonomous navigation system for mobile robots using Deep Reinforcement Learning (DRL). The proposed system allows a robot to explore unknown environments autonomously, identify Points of Interest (POIs), select optimal waypoints, and navigate effectively using a trained motion policy. Unlike traditional approaches, this system requires no prior knowledge or pre-existing maps, dynamically updating its navigation strategy based on real-time sensor data. Through experiments conducted in diverse simulated and real-world scenarios, the system demonstrated superior efficiency, robustness, and adaptability, successfully addressing challenges like local optima and dynamic obstacles.

1 Introduction

Autonomous navigation is a cornerstone of mobile robotics, with applications ranging from search and rescue operations to industrial automation. The ability to explore unknown environments without relying on pre-mapped data is crucial in hazardous, inaccessible, or dynamic scenarios where manual intervention is impractical.

Traditional approaches rely heavily on pre-existing maps and static environments, which limit their flexibility. In contrast, this project develops a hybrid navigation system integrating Deep Reinforcement Learning (DRL) with heuristic-driven global strategies. The objective is to enable robust, real-time decision-making and navigation in unknown environments, leveraging sensor data for dynamic adaptability.

2 Problem Statement

The primary challenge addressed in this project is enabling mobile robots to navigate autonomously in unknown environments toward predefined goals. This involves overcoming obstacles, avoiding collisions, and optimizing path planning in real time without human intervention or pre-existing maps.

Such capabilities are critical for various applications, including:

- **Search and Rescue:** Navigating through disaster zones to locate survivors.
- **Industrial Automation:** Streamlining operations in unstructured warehouse environments.
- **Smart City Navigation:** Guiding autonomous vehicles in urban settings with dynamic obstacles.

By addressing this challenge, the project aims to improve operational efficiency, reduce costs, and enhance safety in these domains.

3 Proposed Approach

The proposed system integrates global and local navigation strategies to achieve goal-driven autonomy:

3.1 Global Navigation

Global navigation focuses on selecting waypoints that guide the robot toward the goal. This process involves:

- Identifying **Points of Interest (POIs)** using sensor data such as LiDAR or cameras.
- Evaluating and scoring waypoints using a heuristic function that considers factors like distance to the goal, obstacle density, and navigability.

3.2 Local Navigation

Local navigation ensures smooth and safe motion between waypoints:

- A Twin Delayed Deep Deterministic Policy Gradient (TD3)-based DRL model generates linear and angular velocity commands.
- The model is trained to prioritize obstacle avoidance and directional alignment while maintaining efficient progress toward the goal.

3.3 Dynamic Mapping

The robot constructs and updates an occupancy grid map of its environment in real time, using sensor readings to detect obstacles and free space. This map aids in navigation and prevents revisiting previously explored areas.

3.4 Rationale

Traditional navigation methods often fail in dynamic or cluttered environments due to their dependence on static maps. The proposed hybrid approach leverages the adaptability of DRL for local navigation and the structured decision-making of heuristic-driven strategies for global planning. This combination addresses challenges like:

- **Local Optima:** Avoiding suboptimal paths caused by immediate obstacles.
- **Dynamic Obstacles:** Adapting to changes in the environment, such as moving objects.
- **Scalability:** Extending the system to larger, more complex environments.

4 Technical Background

4.1 Deep Reinforcement Learning (DRL)

In DRL, an agent interacts with an environment to learn an optimal policy that maximizes a cumulative reward:

$$J(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right],$$

where $\gamma \in [0, 1]$ is the discount factor, balancing immediate and future rewards.

4.2 Policy Gradient Methods

Policy gradient methods optimize a policy $\pi_\theta(a|s)$ by maximizing the expected return:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a_t|s_t) \cdot G_t\right].$$

- **Actor:** Learns the policy for selecting actions.
- **Critic:** Evaluates the action-value function $Q(s_t, a_t)$ to guide the actor.

4.3 Twin Delayed Deep Deterministic Policy Gradient (TD3)

TD3 enhances DRL stability in continuous action spaces by introducing:

- **Twin Critics:** Reduces overestimation bias by maintaining two Q-functions.
- **Delayed Updates:** Updates the actor and target networks at a slower rate.
- **Target Noise:** Adds Gaussian noise to smooth target values.

5 Mobile Robot Kinematics

The robot's motion is governed by its kinematics:

5.1 State Variables

- x, y: Position.
- θ: Orientation.

5.2 Control Inputs

- v: Linear velocity.
- ω: Angular velocity.

5.3 Motion Equations

$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = \omega.$$

Discrete-time model:

$$x_{t+1} = x_t + v_t\cos\theta_t\Delta t, \quad y_{t+1} = y_t + v_t\sin\theta_t\Delta t, \quad \theta_{t+1} = \theta_t + \omega_t\Delta t.$$

6 Reward Function

The reward function encourages desired behaviors:

- **Goal Proximity:** $r_{\text{goal}} = -\alpha d$, where d is the distance to the goal.
- **Collision Penalty:**

$$r_{\text{collision}} = \begin{cases} -\beta & \text{if collision detected,} \\ 0 & \text{otherwise.} \end{cases}$$

- **Heading Alignment:** $r_{\text{align}} = -\gamma |e_\theta|$, where e_θ is the heading error.

7 Experiment Setup

7.1 Environment

- **Simulated:** 10x10m space with randomized obstacles in Gazebo11.

7.2 Implementation

- **Software:** ROS2, Python3.8, Gazebo11, Rviz.
- **Hardware:** NVIDIA GTX 1650Ti, Intel i7-6800K.

8 Results and Discussion

The robot demonstrated smooth and adaptive navigation through diverse environments, seamlessly integrating global and local strategies. It moved efficiently toward goals by dynamically selecting waypoints and adjusting its path in response to real-time sensor data. The TD3-based motion policy enabled precise control of linear and angular velocities, ensuring collision-free movement even in cluttered and dynamic settings. The robot exhibited an ability to navigate tight spaces, avoid obstacles, and maintain a heading aligned with the goal, reflecting the robustness of the learned policy. Overall, the robot's movements were fluid and goal-oriented, balancing speed and safety to achieve efficient exploration and navigation. The system achieved significant improvements in:

- **Efficiency:** Reduced navigation time.
- **Reliability:** Consistently reached goals across diverse scenarios.
- **Robustness:** Effectively handled dynamic obstacles and clutter.

Challenges include scalability and hardware dependencies.

9 Conclusion

This project successfully developed a goal-driven autonomous navigation system integrating TD3-based DRL with heuristic strategies. Future work will focus on enhancing scalability and adaptability for broader applications.