Project Proposal: CUDA-Accelerated Reinforcement Learning for Route Optimization

Hiruy Worku

Route optimization is a fundamental problem in logistics, transportation, and navigation systems, where the goal is to find the most efficient path between locations. Traditional methods, such as Dijkstra's algorithm and A*(A-Star), provide optimal solutions but can be computationally expensive for large-scale environments. Reinforcement Learning (RL) offers a promising alternative by enabling AI agents to learn optimal routing strategies through interaction with a dynamic environment.

Several studies have explored the benefits of reinforcement learning in navigation and optimization problems. Prior research has demonstrated the effectiveness of RL-based approaches in dynamic traffic systems, autonomous vehicles, and smart city planning. Recent advances in deep reinforcement learning, particularly with policy optimization methods like Proximal Policy Optimization (PPO) and Deep Q-Networks (DQN), have shown significant improvements in pathfinding efficiency and adaptability compared to traditional static algorithms. Additionally, the advent of CUDA-powered GPU acceleration has enabled faster training and execution, making RL more feasible for real-time applications.

This project aims to develop a reinforcement learning model for route optimization, utilizing existing RL libraries such as Stable-Baselines3 and OpenAI Gym. The model will be trained to navigate a simulated grid-based campus map, where it must determine the shortest and most efficient routes while avoiding obstacles. The primary objective is to compare training performance and decision-making accuracy between CPU-based and CUDA-accelerated implementations using PyTorch.

The project consists of two key components. First, a custom OpenAI Gym environment will be created to simulate different route optimization scenarios, including traffic conditions and roadblocks. The agent will interact with the environment, receiving rewards based on travel efficiency and penalty for unnecessary detours. Second, a reinforcement learning model (e.g., PPO or DQN) will be trained using Stable-Baselines3 with CUDA acceleration, optimizing key computations such as policy updates and matrix operations. A performance comparison between CPU and GPU execution will be conducted to assess training efficiency and final model accuracy.

A computer drawn prototype illustrating the route optimization process, agent decision-making, and system architecture is attached. This visualization outlines how the RL agent will interact with the environment and learn optimal routing strategies over time.

As a stretch goal, real-world map data integration using the Google Maps API may be explored to test the model's applicability to real-world navigation. The findings from this project will provide insights into the efficiency of RL-based routing methods compared to traditional algorithms and contribute to ongoing research in AI-driven path optimization.

References

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.

[2] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, and D. Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *Nature*, 550(7676):354–359, 2017.

[3] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 1998.

Appendix: Software Outline

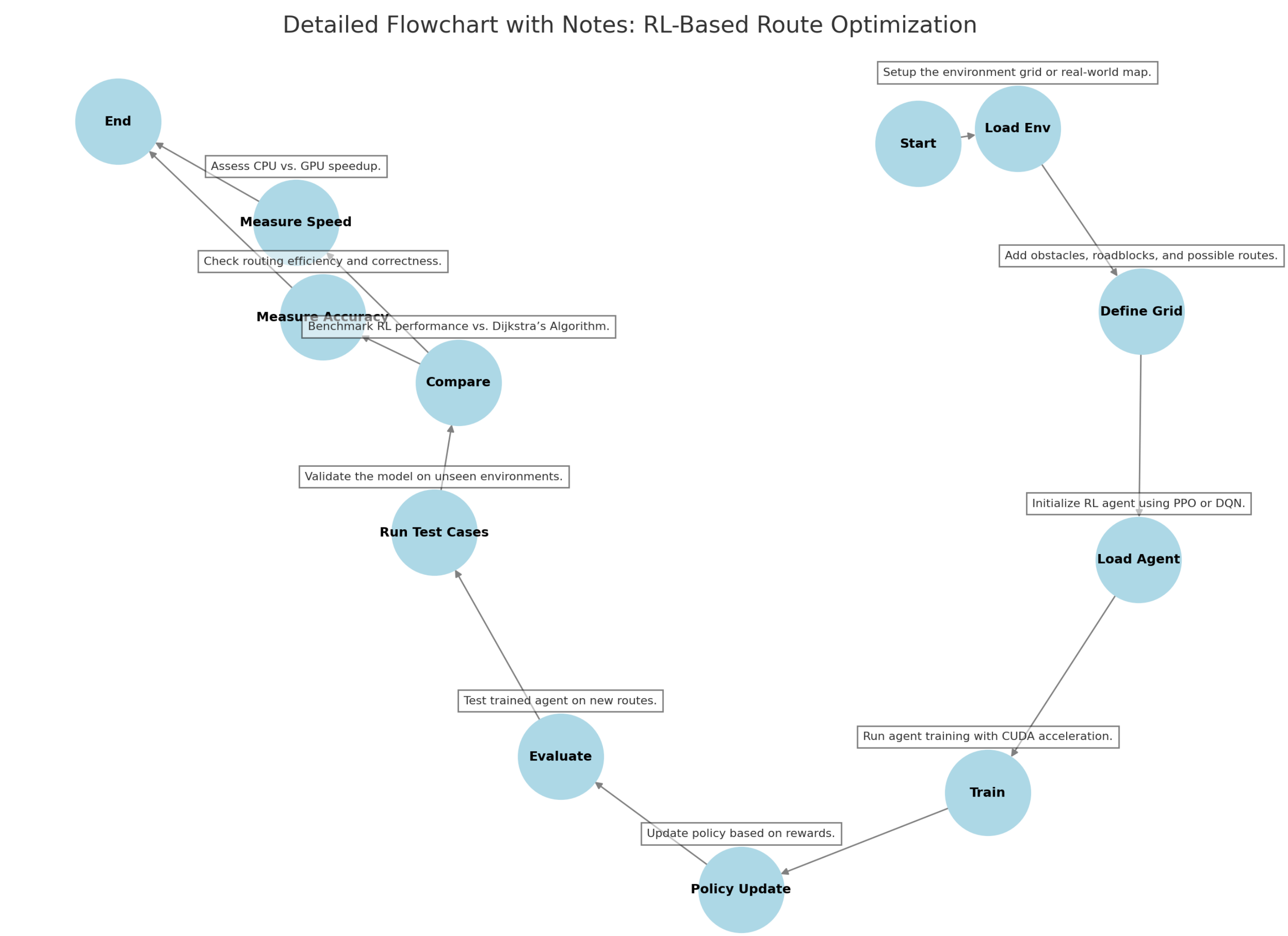

Figure 1: System Architecture for RL-Based Route Optimization

Appendix: Agent interacts with the environment

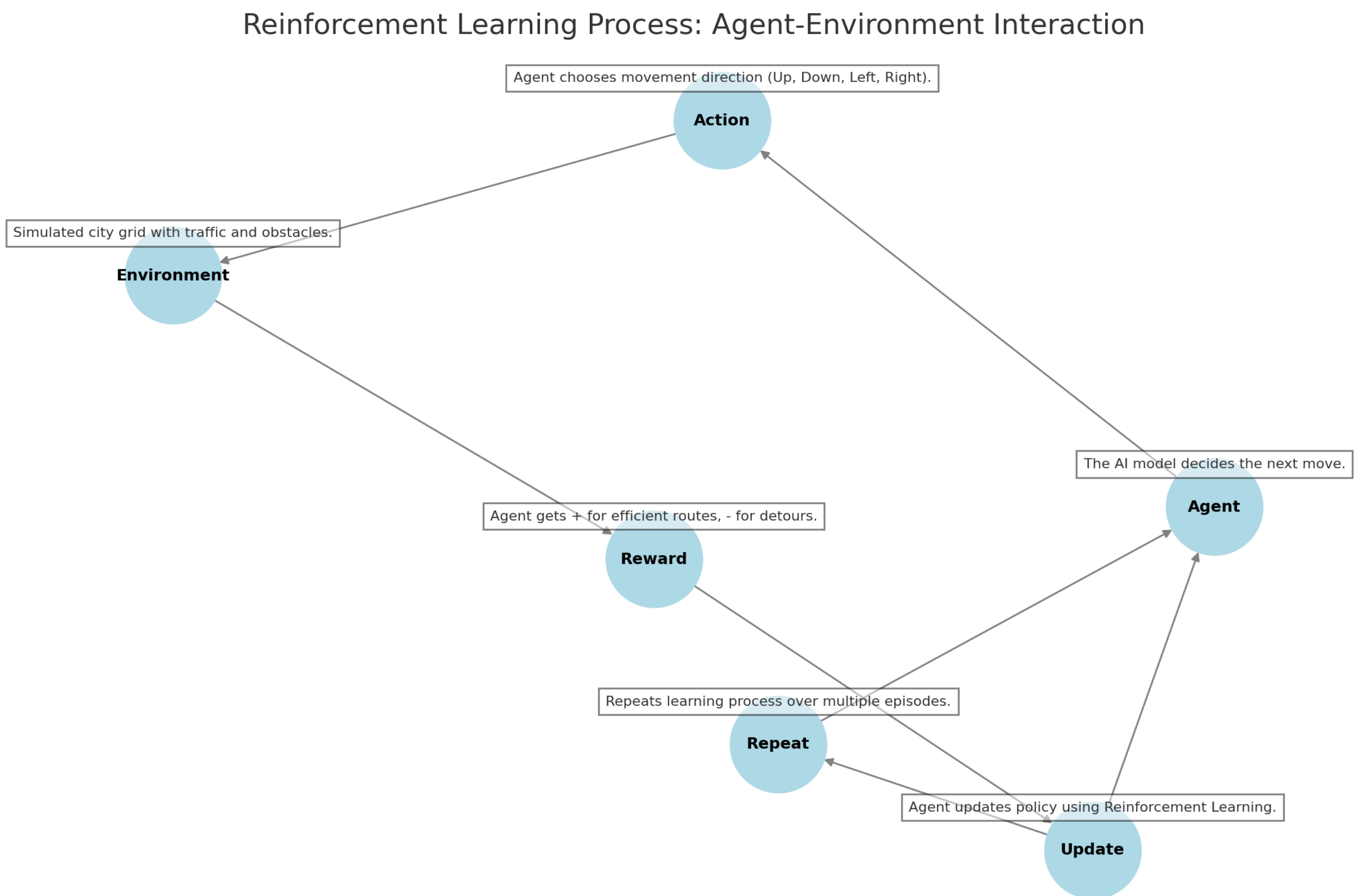

Figure 2: System Architecture for RL-Based Route Optimization