Submission and Formatting Instructions for International Conference on Machine Learning (ICML 2022)

Anonymous Authors[1]

Abstract

This report presents a study on supply chain management using Deep Q-Network (DQN) reinforcement learning. We investigate the performance of a single-agent DQN baseline algorithm in optimizing the profit of a specific firm within a multi-firm supply chain environment. The environment simulates a supply chain with multiple firms, where each firm's demand is generated using a Poisson distribution. The DQN agent is trained to maximize its profit by making optimal ordering decisions, while other firms follow either random or classical supply chain management strategies. Our results demonstrate the effectiveness of the DQN approach in improving the profit of the trained firm compared to traditional strategies.

1. Introduction

Supply chain management is a critical area in operations research and management science. The ability to optimize ordering decisions can significantly impact a firm's profitability. In this study, we explore the application of reinforcement learning, specifically Deep Q-Network (DQN), to optimize ordering decisions in a supply chain environment. We focus on a single-agent approach where one firm is trained using DQN, while other firms follow predefined strategies.

2. Methodology

2.1. Environment

The supply chain environment is simulated with multiple firms, each characterized by parameters such as price, holding cost, and lost sales cost. The demand for the downstream firm is generated using a Poisson distribution, while the demand for upstream firms is determined by the orders placed by downstream firms.

2.2. Deep Q-Network (DQN)

The DQN algorithm is used to train a single agent (firm) to make optimal ordering decisions. The agent's state space includes the current order, satisfied demand, and inventory levels. The action space is discrete, representing the order quantity. The reward is defined as the profit, which is calculated based on the revenue from satisfied demand, the cost of orders, and the holding cost of inventory.

2.3. Training Process

The training process involves multiple episodes, where each episode consists of a series of steps. The agent interacts with the environment, observes the state, takes an action, receives a reward, and updates its policy using the DQN algorithm. The exploration rate is gradually decreased to balance exploration and exploitation.

3. Results

3.1. Training Performance

The training performance is evaluated based on the cumulative reward (profit) achieved by the DQN agent over episodes. The results show that the DQN agent improves its performance over time, achieving higher profits compared to random strategies.

3.2. Testing Performance

The testing performance is evaluated by running the trained agent in the environment for a fixed number of episodes. The results demonstrate the agent's ability to make optimal ordering decisions, leading to higher profits compared to other firms following random or classical strategies.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

4. Discussion

The results indicate that the DQN approach is effective in optimizing the profit of the trained firm. The agent learns to balance the trade-off between ordering too much (increasing holding costs) and ordering too little (increasing lost sales costs). However, the performance of the DQN agent may vary depending on the strategies followed by other firms in the supply chain.

5. Conclusion

In conclusion, the application of DQN in supply chain management shows promising results in optimizing the profit of a single firm. Future work could explore multi-agent DQN approaches to optimize the entire supply chain's performance.

6. Future Work

Future research could focus on:

- Extending the DQN approach to multi-agent scenarios.
- Investigating the impact of different demand distributions on the performance of the DQN agent.
- Exploring the use of other reinforcement learning algorithms, such as Policy Gradient or Actor-Critic methods.

A. You can have an appendix here.

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one, even using the one-column format.