

Multi Agent Reinforcement Learning Based Warehouse Task Assignment

K.A. Ashan Priyadarshana

218537D

MSc/PG Diploma in Artificial Intelligence

Department of Computational Mathematics

Faculty of Information Technology

University of Moratuwa

Sri Lanka

April 2024

Multi Agent Reinforcement Learning Based Warehouse Task Assignment

K.A. Ashan Priyadarshana

218537D

Thesis/Dissertation submitted in partial fulfillment of the requirements for the
degree

MSc/PG Diploma in Artificial Intelligence

Department of Computational Mathematics

Faculty of Information Technology

University of Moratuwa

Sri Lanka

April 2024

DECLARATION

I declare that this is my own work and this thesis/dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other University or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature

Date:

The above candidate has carried out research for the PhD/MPhil/Master's thesis/dissertation under my supervision. I confirm that the declaration made above by the student is true and correct.

Name of Supervisor: Prof. Asoka S. Karunananda

Signature of the Supervisor:

Date:

DEDICATION

To my beloved parents, S.A Pathmawathi and K.A Asoka Priyantha. Your love, endless support, and unwavering belief in me have been the foundation of my life and my achievements. This work is a testament to your guidance and the importance you instilled in education.

ACKNOWLEDGEMENT

I would like to express my profound gratitude to my thesis supervisor, Professor Asoka S. Karunananda. From the very beginning of my master's program, his ability to inspire a fresh perspective has been transformative. His lectures were consistently engaging, and his guidance throughout my research was invaluable. I deeply appreciate his wisdom, support, and the positive impact he has had on my academic journey.

ABSTRACT

The growth of e-commerce and the demand for rapid fulfillment place a high pressure on modern warehouses to optimize efficiency. Central to this problem is the dynamic allocation of different tasks to a workforce comprised of both humans and robots. The task allocation system must prioritize time-sensitive picking tasks and ensure the timely completion of crucial supporting tasks like replenishment, all while considering the availability of specialized devices required for different task types. Conventional rule-based or heuristic systems struggle to adapt and optimize decision-making within such a complex operating environment.

Recent literature shows interest in applying reinforcement learning for warehouse management, including task allocation. Reinforcement Learning algorithms are well-suited to environments with clear rewards and observable states. However, most existing work focuses on single-agent scenarios or limited aspects of warehouse operations. In contrast, real-world warehouse task allocation inherently involves a cooperative multi-agent setting.

This thesis adopts a Multi-Agent Reinforcement Learning (MARL) framework to address the challenge of optimizing task assignment within a warehouse environment characterized by a workforce comprised of both human workers and autonomous robots. A central challenge addressed in this framework is the need to prioritize and allocate tasks with varying levels of urgency while considering the distinct constraints and capabilities associated with different worker types. To capture the dynamic nature and interdependencies of warehouse tasks, a comprehensive representation is used. Each task is defined by a set of attributes including its type (e.g., 'pick', 'replenish', 'load', or 'put'), the specific product or Stock Keeping Unit (SKU) involved, quantity, source and destination locations, timestamp, and order association. Notably, pick tasks are directly linked to specific customer orders for prioritization, while replenishment tasks may be indirectly related if they serve to stock locations for picking operations. Additionally, a 'status' attribute indicates whether the task is 'available', 'active', or 'completed'. Devices essential for task completion are modeled with attributes indicating their type (e.g., forklifts and pallet jacks), availability status, and any currently assigned tasks. The MARL system leverages agents that learn effective decision-making based on the current state of the warehouse. This state information is composed of two key elements provided as input to each agent's Deep Q-Network (DQN). The first element is the task state, which includes the current catalog of available tasks along with their detailed attributes. Secondly, device state information is fed into the DQN, providing details about the availability of devices and their current associations with tasks, if any. By considering both task information and device availability, the agents are equipped to make informed decisions for optimal task allocation.

A carefully designed reward mechanism is crucial to guide agents towards optimal task selection and resource utilization. The reward structure is designed to reinforce the timely completion of customer orders, prioritizing pick tasks. To ensure uninterrupted picking workflows, replenishment tasks necessary to facilitate pick operations receive a secondary level of reward. Overall warehouse efficiency is encouraged through a base-level reward for the completion of other tasks. Importantly, negative rewards are introduced as an obstacle against actions that lead to wasted

resources or unnecessary delays. This includes actions such as selecting tasks that are already in progress, attempting to use unavailable devices, or remaining idle.

At the core of the proposed framework is a Multi Agent Reinforcement Learning approach utilizing Deep Q-Networks (DQNs) as the decision-making policy for each agent. DQNs, a powerful form of reinforcement learning, allowing agents to learn the value of different action within a given state. Over time, as agents interact with the environment and observe the outcomes of their decisions, they learn to make optimal choices. The use of individual DQNs for each agent, both human and robotic, is critical. This individualized DQN approach allows for personalized learning plans tailored to the specific capabilities of agent type, making the overall task allocation system more effective and adaptable.

The framework is evaluated through a series of simulation experiments. Scenarios are designed to assess the MARL-based system against a baseline random policy, focusing on assessing the system's impact with a human-only workforce and limited devices and examining the system's performance and adaptability with the introduction of robots alongside human workers.

Simulation results demonstrate a consistent improvement in key efficiency metrics when more agents employ the learned DQN policy. Notably, pick task completion efficiency shows substantial improvements, indicating the framework's success in prioritizing time-critical outbound tasks. Furthermore, the MARL system demonstrates effective device utilization, with a decrease in idle resources. This thesis confirms MARL's potential for optimizing task allocation in warehouse environments with heterogeneous workforces. Future extensions could explore more sophisticated coordination techniques across agents and incorporate warehouse layout into the decision-making process.

TABLE OF CONTENTS

DECLARATION.....	i
DEDICATION	ii
ACKNOWLEDGEMENT	iii
Abstract.....	iv
TABLE OF CONTENTS	vi
LIST OF FIGURES	x
CHAPTER 1	1
INTRODUCTION	1
1.1 Prolegomena	1
1.2 Objectives	1
1.3 Background and Motivation.....	1
1.4 Problem in Brief.....	3
1.5 A Novel Approach to Warehouse Task Assignment	3
1.6 Resource requirements.....	4
1.7 Structure of the Thesis	4
1.8 Summary	4
Chapter 2.....	5
Developments and Challenges in Warehouse Taks Assignment	5
2.1 Introduction.....	5
2.2 Technological Advancements in Warehouse Management	5
2.3 Impact on Task Allocation and Gaps	5
2.4 Task Assignment as an Optimization Problem.....	5
2.5 Simulation-Based Optimization for Worker Assignment.....	6
2.6 Automated storage and retrieval systems (AS/RS)	8
2.7 Multi Agent systems in real-world warehouses	9
2.8 Warehouse Management Benchmarks	11
2.9 Overview of Developments and challenges in the warehouse task assignment	11
2.10 Problem definition	12
3.11 Summary	12
Chapter 3.....	14
Theoretical foundations of Reinforcement Learning and Multi Agent Systems.....	14
3.1 Introduction.....	14

3.2 Simulation-Based Optimizations	14
3.3 Genetic Algorithms for Warehouse Task Assignment.....	14
3.4 Multi-Agent Path Finding (MAPF) within Warehouses	15
3.5 Policy Gradients and Actor Critics for Warehouse Operations	15
3.6 Case-Based Reinforcement Learning.....	16
3.7 Hierarchical MARL for Order Picking	17
3.8 Zero-Shot Task Generalization	17
3.9 Cooperative Multi-Agent Reinforcement Learning for Efficient Inventory Management	18
3.10 Use of Heuristic Algorithms in Warehouses	18
3.11 Use of Multi-Agent Reinforcement Learning for Warehouses	19
3.12 A Technical Overview of MARL	20
3.12.1 Key Areas Under Multi-Agent Reinforcement Learning	20
3.12.2 Use of Deep Q Networks in MARL for Task Assignment Problem	21
3.13 Summary	21
Chapter 4.....	23
An Approach to Warehouse Task Assignment Using MARL	23
4.1 Introduction.....	23
4.2 Hypothesis.....	23
4.3 Inputs	23
4.4 Output	23
4.5 Process	24
4.6 Users	24
4.6 Features	24
4.8 Summary	25
Chapter 5.....	26
MARL Based Design for Task Allocation in Hybrid Warehouses	26
5.1 Introduction.....	26
5.2 Top-level architecture.....	26
5.3 Setup Module	27
5.4 Tasks Module	28
5.5 Device Module.....	28
5.6 Reward Module.....	28
5.7 KPI Calculation Module	28
5.8 Action Module	28
5.9 Observation Module.....	29

5.10 Agent.....	29
5.11 DQN (Deep Q Network)	29
5.8 Summary	29
Chapter 6.....	30
Implementation of MARL Based Warehouse Task Assignment	30
6.1 Introduction.....	30
6.2 Setup Module Implementation.....	30
6.3 Tasks and Device Module Implementation.....	30
6.5 Reward Module Implementation	31
6.6 KPI Module Implementation	31
6.7 Action and Observation Module Implementation	32
6.9 Agent and DQN Implementation	33
6.10 Summary	33
Chapter 7.....	34
Evaluation	34
7.1 Introduction.....	34
7.2 Evaluation Strategy	34
7.3 Experimental Setup.....	34
7.4 Results.....	36
7.4.1 Results of Simulation 1	36
7.4.2 Results of Simulation 2	38
7.4.3 Results of Simulation 3	40
Chapter 8.....	43
Conclusion and Further Work	43
8.1 Introduction.....	43
8.2 Overall conclusion	43
8.3 objective-wise achievements.....	43
8.4 Limitation.....	44
8.5 Further work.....	44
8.6 Summary	45
References.....	46
Appendix A	49
An Introduction to Multi-Agent Systems.....	49
Agent Architectures	49
Communication and Coordination	50

Cooperation and Negotiation	51
Distributed Problem Solving.....	51
An Introduction to Reinforcement Learning.....	52
Markov Decision Processes (MDPs)	52
Value Functions.....	53
Bellman Equation.....	54
Policy Based Methods.....	54
Value-Based Methods	55
An Analogy for Policy and Value Based Methods.....	55
Exploration vs Exploitation	55
Model-Based Learning vs Model-Free Learning.....	56
An Analogy	57
Appendix B	58
The Code for Model Creation	58
Description.....	58

LIST OF FIGURES

Figure	Description	Page
Figure 5.1	MARL based design for warehouse task assignment	26
Figure 7.1	Setting 1 – Scenario 1 Task Completion Chart	35
Figure 7.2	Setting 1 – Scenario 2 Task Completion Chart	36
Figure 7.3	Setting 1 – Scenario 3 Task Completion Chart	36
Figure 7.4	Setting 2 – Scenario 1 Task Completion Chart	37
Figure 7.5	Setting 2 – Scenario 2 Task Completion Chart	38
Figure 7.6	Setting 2 – Scenario 3 Task Completion Chart	38
Figure 7.7	Setting 3 – Scenario 1 Task Completion Chart	39
Figure 7.8	Setting 3 – Scenario 2 Task Completion Chart	40
Figure 7.9	Setting 3 – Scenario 3 Task Completion Chart	40
Figure 9.10	Developed Virtual Environment Using Gymnasium and pygame Libraries	42
Figure 8.1	Pick Task Completion Efficiency Summary	43
Figure B.1	DQN Model Implementation	57