CHAPTER I

INTRODUCTION

1.1 Background

Warehouse industry is one of the continuously growing major industries in the world. With the recent explosive growth of e-commerce, storage and manufacturing system becomes more and more important along with the logistics inside and outside of the warehouses themselves. Along with it, automation also has become the subject of interest in the recent years and achieved substantial growth in the scene. Thus, the subject of automation in storage and manufacturing industries becomes one of the topics that would have an impact on how our industries would improve and expand. The current state of the integration of various automation is recognized as Industry 4.0 [1]. One of the subject areas of automation in warehouses is their transport system. Automation has been done in the form of vehicles with the capability to transfer objects from point A to point B. Said vehicles can typically be recognized as Automatic Guided Vehicles (AGV) as seen on Figure 1.1.

Figure 1.1. Forklift AGV with Stabilizer Pad

(https://en.wikipedia.org/wiki/Automated_guided_vehicle)

While AGVs continuously been improved, advancements have been made in the scene and Autonomous Mobile Robot (AMR) becomes part of the industry [2],[3]. While AMR itself is not limited to the scope of logistics, AMR has increasingly become a more popular as a subject of interest. Thus, AMRs in the form of warehouse related vehicles has been increasingly making more appearance in the industry.

The existence of AMR in the industry would significantly reduce the need for potentially dangerous labour as it can be potentially used for various purposes such as material transport, inventory management and hazardous materials handling. AMRs can also have higher efficiency and productivity given the massive reduction in manual labour and 24/7 automated operations. While the cost of initial investment might be relatively high, the usage of AMRs may outweigh risk and costs in long-term conditions [2], [16] - [18]. AMR themselves have various programming components in them and while every component plays an important part in harmonious combination, navigation and path planning remains the basics of the system. Thus, understanding the task allocation and pathfinding algorithm can be considered as one of the main topics in AMR to be discussed and dissected [1],[3].

Pathfinding can be divided into two categories depending on the environment which are uninformed search and informed search. Uninformed search typically works on domains that has no known previous information and focuses on generates on how the domain would look like instead of creating a more optimized pathway. Informed search on the other hand, already has information provided into their systems. This indicates that a storage building layout has been given pre-emptively before creating a pathway for the AMRs to traverse.

AMRs may use a hybrid system approach where it would use both uninformed and informed search with both methods are done for different contexts. AMRs may use uninformed search as the initial stage with the purpose of mapping. Once the map has been created, human operators may manually label points or areas in the map that represent storage spaces or various other applicable labels. Labelled points would then be considered as various goal points for the AMRs, and informed pathfinding algorithms take their share of the work as it would create a more optimal pathway from entry to goal points. While AMRs work does not stop at just pathfinding, this would work as one of the least sensitive components of AMRs. The possibility of entirely skipping the mapping process can also be done if map layouts are provided, well-defined and static. However, applying dynamic path planning is still a necessity as there are various factors in the element that may cause changes in the domain. One of such important elements is the existence of multiple vehicles operating in the same domain within the same time frame. The existence of multiple AMRs requires a more complex algorithm. The approach of having multiple AMRs, referred as agents, is called Multi Agent Path Finding (MAPF). MAPF problems expands the task allocation to multiple agents. MAPF task allocation can be decentralized or centralized. The visual representation of either approach can be seen on Figure 1.2.

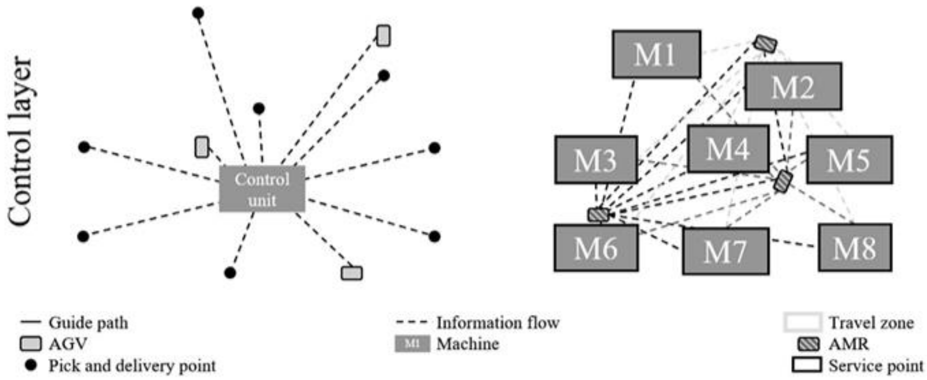

Figure 1.2. Centralized Approach (Left) and Decentralized Approach (Right) [4]

Both approaches are active field of study that's been researched, studied, and continuously improved as more and more advances are being made [1],[8].

After the consideration of the task allocations, pathfinding algorithms in the form of MAPF would then be needed as an attempt to complete the assigned task for the AMRs [5]. MAPF is specifically used due to the nature of multiple agents are actively searching for solutions which would attempt to avoid possible collisions with one another [6],[7]. There are a lot of different pathfinding algorithms that has been found and used, but in terms of popularity, some of the more known ones are Conflict-Based Search (CBS), Enhanced Conflict-Based Search (ECBS), and other variants of A* such as Cooperative A* (CoopA*) [7]. It is also important to recognize conflicts in MAPF in which conflicts are situations when different agents attempt to occupy the same space [6]. The pathfinding algorithms mentions are known to be informed search algorithms as the significance of the continuity of the operation plays further role than the initial stage of mapping in which only acts as the initial domain inspection. The efficiency of respective algorithms may vary depending on the context as every algorithm typically have their own strength and weaknesses. With this consideration, then selecting the appropriate pathfinding algorithm for different circumstances becomes important.

Considering the given process, there is a need to analyse the amount of computational task done in the context of navigation. The number of agents and complexity influence the amount of computational task as the more agents there are, the more processes are needed [1],[4],[7]. The computational task does not only stop on pathfinding process as there is still a need in addressing collision and other various dynamic changes that may happen. With such consideration, the amount of computational task might skyrocket and lead to significant delays if incorrect approaches are used.

Thus, given all the previous conditions, the question is how to reduce computational resources when high numbers of AMRs with different task assignments are used in warehouse environments. Seeking possible model solution is the motivation for the research described in this thesis.

A proposed hybrid model which uses Conflict Based Search (CBS) and Multi Agent Reinforcement Learning (MARL) is used by combining CBS rule-based algorithm as primary high-level pathfinding algorithm and sends the path generated to MARL as navigation algorithm for actual navigation amongst agents preventing collisions with one another.

1.2 Problem Identification

Navigation in autonomous vehicles is a prevalent problem that currently have many proposed solutions to the problem. This includes within the context of warehouse industries where AGV and AMR are regularly used. However, various solutions of the problem typically revolve around a single concept of either rule-based method or a learning method. A rule-based method known as Conflict Based Search (CBS) is one of the more popular algorithms in understanding navigations in autonomous vehicles, in which warehouse environment are one of the easier environments the algorithm can be used on due to the common nature of warehouse to be in a grid like system. CBS can be applied with different levels of strictness. When the restrictions are set on a loose condition, there are various security concern that needs to be addressed. At the same time, when restrictions are too tight, the resource use can be significantly higher. Understanding the trade-off within the CBS remains as an important part of using the algorithm as is. The problem then lies on the nature of complexity in the form of agent numbers where multiple agents may cause significant computing resource use upon task execution leading to inefficient and potentially failure in the system as restrictions are set tightly. The same complexity can also cause catastrophic failure when collision concerns within the agents are not addressed carefully when restrictions are loose. While there are various enhancements made in an attempt to solve the problem, majority of the solutions are still within the scope of the same rule-based method. We propose to use Multi Agent Reinforcement Learning (MARL) as secondary pathfinding algorithm to support continuous space movements. MARL's nature as a learning algorithm as opposed to rule-based algorithm requires MARL to be trained. MARL as a learning algorithm have the weakness of resource needed for training leading to scalability issues [28],[39].

Considering the advantages and disadvantages of both methods, there is a possibility of leveraging both the strength of a rule-based method and learning method where CBS can act as a high-level pathfinding algorithm where loose restriction for CBS is used, and Multi Agent Reinforcement Learning (MARL) is used as a low-level pathfinding algorithm in an attempt to solve the collision concerns CBS may have. The hybridization of the algorithm may provide a solution to reduce resource use during both training and execution while assuring safety on the agents' movements.

1.3 Problem Limitations

This research will assume that the environment will be made as close as possible to various generic layouts of a warehouse. The environment is also made on a grid-based basis where obstacles or shelves are considered as squares. The reasoning behind this is objects within the warehouse environment are typically placed within pallets or boxes which are rectangular. The racking frame placed within the warehouse are also typically in a rectangular shape. This would mean that while perfection may not be achieved, squares are somewhat common practice to be used as grids and are within acceptable tolerance.

The agents are also considered to have no friction and no complex mechanisms such as torque, differential, et cetera which a typical vehicle would have. Agents are also assumed to only have front-facing movement and are to only move in a two-dimensional plane. This is to simplify the agents as the main goal is the general navigation instead of the mechanical capacity of an agent. This would mean that various movements are accepted to be possible, disregarding the mechanical limitations an actual AMR may have. Movement will be taken as discrete actions instead of continuous to provide a more simplistic approach while still providing acceptable agent actions. The discrete movements assigned are:

1. Moving in a constant speed
2. Turning 5 degrees to the left
3. Turning 10 degrees to the left
4. Turning 5 degrees to the right
5. Turning 10 degrees to the right
6. Stopping

The experiments are done by assigning 4 different cases of agent numbers which are 5 agents, 10 agents, 15 agents, and 20 agents.

1.4 Problem Definition

To build the hybrid model as explained, it is required to solve the ongoing problems as follows:

a) How to recreate an environment suitable and representative of a warehouse environment?
b) How to address the trade-off problems encountered by CBS algorithms in favour of good stability while maintaining safety?
c) How to build a learning algorithm to patch the gaps on applying CBS as a rule-based algorithm?
d) How to evaluate the performance the use of the proposed model in comparison to a strict CBS model within the same environment?

1.5 Research Purpose

The use of CBS algorithm in the context of autonomous vehicles may provide unsatisfactory results when introducing various trade-offs. One of the considered trade-offs is how strict the CBS algorithm will be to ensure safety. A highly strict CBS is more likely to ensure safety by maintaining zero tolerance in the pathfinding algorithm. However, as more complex environment is introduced towards the system, it is highly likely it will also take much higher computational power during execution thus leading to what can be considered as a failure in execution. This thesis attempts to create optimization by introducing a hybrid model using CBS and Multi-Agent Reinforcement Learning (MARL). The CBS, used as high-level pathfinding algorithm, will be more lenient allowing various tolerances in which low-level algorithm will patch the gap created by the CBS trade-off. The proposed model aims to reduce computational load in comparison to a standard CBS model. The success standards are based on the capability of the proposed model to complete tasks within assigned evaluation standards. The proposed model is designed to achieve the desired performance with minimal degradation, even as more agents are introduced into the environment, compared to the base CBS algorithm.

1.6 Outline of the Thesis

This research consists of 5 chapters. Every chapter explains its content within a specific scope with a systematic schema of understanding. The chapters can be explained as follows:

Chapter 1 Introduction contains brief introduction in understanding the current problem that autonomous vehicles, specifically within the warehouse industry, may encounter. One of the many problems that is highlighted in the chapter is the issue of computational resource use when the system is introduced to multiple agents and the security issues that ensues. The chapter also briefly explains the aim of creating the proposed method while narrowing the research with simulated limitations.

Chapter II Theoretical Background explains the history and theories that are relevant to the research. This chapter covers the basic understanding of AGVs and AMRs, Centralized and Decentralized task assignments, and Multi-Agent Path Finding (MAPF) Problems. This chapter is then proceeded with a more in-depth theory of the algorithms that will be used in the research which are Conflict-Based Search (CBS) algorithm as high-level pathfinding algorithm, Reinforcement Learning (RL) as single agent low-level pathfinding algorithm, and Multi-Agent Reinforcement Learning (MARL) as a further extension of RL when introduced to a multi-agent system. This chapter is the accumulation of theories to determine the proposed research methodology in the next chapter.

Chapter III Research Methodology provides the details of the research planning and experiments process. A block diagram of the experiment process is provided in this chapter as a visualization tool for the experiment workflow. In general, the process is divided into 4 phases which are high-level pathfinding using CBS, low-level pathfinding using RL, multi agent training, and evaluation. Low-level algorithm ensures to create agent trails that can be used for low-level algorithm by using CBS with less conflict constraints. The low-level algorithm is then responsible to address the collision risk the previous algorithm has by navigating in continuous space rather than grid-based environment.

Chapter IV Result and Discussion will explain the observation results of the experiments and provide a commentary of the relevant impact the experiment leads on.

Chapter V Conclusion and Suggestion is the summarization of the research based on the evaluation results in the previous chapter. Suggestions are then provided as further potential research to achieve both better results, and better likeliness to real-life conditions.