Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput

By

Khoo Zong Zheng

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Token-Based Path Allocation Approach For Mobile Robots in Sorting Centre For Achieving High Throughput

Academic Session: Jan 2024

I KHOO ZONG ZHENG
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature) (Supervisor's signature)

Address:
19, Jalan Raja Abdul Rahman 4
Taman Muhibbah, 42200 Kapar
Selangor

Prof Ts Dr Liew Soung Yue
Supervisor's name

Date: 24/4/2024

Date: 26/4/2024

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 24/4/2024

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ***Khoo Zong Zheng*** (ID No: ***20ACB01908***) has completed this final year project entitled "Token-Based Path Allocation Approach For Mobile Robots in Sorting Centre For Achieving High Throughput" under the supervision of Prof Ts Dr Liew Soung Yue from the Department of Computer Science, Faculty of Information and Communication Technology, and Ts Wong Chee Siang from the Department of Computer Science, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

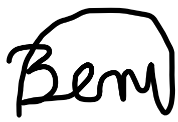
____

(*Khoo Zong Zheng)*

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**METHODOLOGY, CONCEPT AND DESIGN OF A 2-MICRON CMOS DIGITAL BASED TEACHING CHIP USING FULL-CUSTOM DESIGN STYLE**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.



Signature : ____________________________

Name : Khoo Zong Zheng

Date : 24/4/2024

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Dr. Liew Soung Yue who has given me this bright opportunity to engage in this project. Guidance and suggestion gives are much appreciated, a million thanks to you.

I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course. Finally, I would also like to thank my friends for their suggestions and emotional supports that make me possible to finish the whole project in time.

ABSTRACT

As the trend of online shopping and e-commerce rises rapidly after the COVID-19 pandemic, warehouses and sorting centres are facing challenges of managing an enormous number of parcels on a daily basis. To cope with these demands, changing their warehouse to automation are one of the solutions that these companies apply. Autonomous Mobile Robots (AMR) or Automated Guided Vehicle (AGV) is applied to these automated warehouses to sort and deliver these parcels to the desired destination. Normally, these robots are deployed in large quantities. Thus, it is crucial to guaranteed that these robots avoid collisions and deadlocks among each other while executing tasks.

There are some previous approaches that provides different sets of assumptions on the road systems, planning algorithms and layouts. However, some of these assumptions does not obey real-life phenomena such as inertia. Thus, it may be hard to apply this proposed solution in the use of real-life. Also, there are papers that proposes algorithms that are able to solve deadlocks and collision among these robots but not able to solve the heavy traffic problem while passing through an intersection grid cell. Heavy traffic will affect the throughput of completing the tasks. Thus, these problems seem to be the reasons that may be not that applicable and inefficient to apply in the real world.

From here, we will first review and evaluates some of the approaches that have been done by previous studies. These papers will be reviewed to find some inspiration and discover some potential new approaches that might be able to handle multiple robots at the same time while decreasing the traffic as much as possible.

This project proposes an improved path-finding algorithm that implements token-based path finding while a robot wants to access a grid cell. Besides that, the projects also propose to form and break clusters when these robots are moving in a same direction to decrease the queueing time and increases the throughput. Also, a layout is proposed that implements ring road and/or central road concept which allow these robots to travel using these roads without passing the main road that links to the pick-up and drop-off areas.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AGV	Automated Guided Vehicles
AMR	Autonomous Mobile Robots
AS/RS	Automated Storage and Retrieval System
MAPF	Multi Agent Path Finding
MAPD	Multi-Agent Pickup and Delivery
MRS	Multi Robot System
SR	Swarm Robotics
TP	Token Passing
TPTS	Token Passing Task Swaps

CHAPTER 1 INTRODUCTION

Chapter 1 will be discussing about the project background, problem statements, motivations, and objectives. Project background will provide a brief description on the projects and some background information about the topic. For the problem statement, it will show the problem that the project regarding parcel sorting that the world is currently facing. Besides that, the motivations states on the intention that keeping this project going. Lastly, the contribution section will state that how does this project able to benefits the society.

1.1 Background

E-commerce and online shopping are on the rise especially after COVID-19 happens. E-commerce means that using digital methods to sell products and services to customers [1]. With the continuous growth on e-commerce, most of the companies implemented e-commerce strategies for better growth on their business. From the past few years, the trend about E-commerce steadily increases. Retail e-commerce sales worldwide has been increased from 1.3 trillion USD at the year of 2014 to 4.9 trillion USD at the year of 2021 [2].

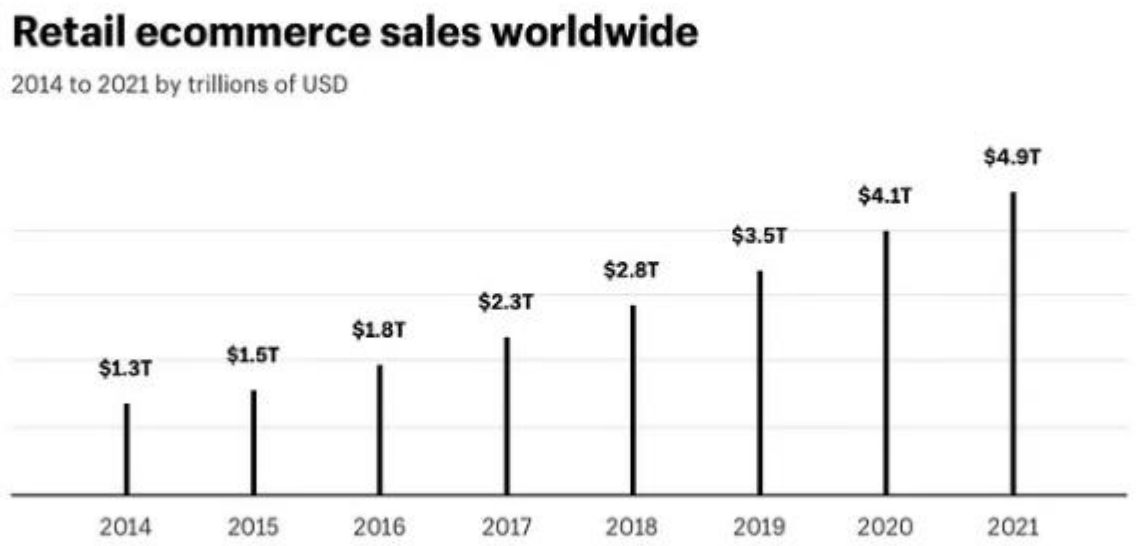


Figure 1 Retail Ecommerce sales worldwide [2]

E-commerce includes selling services or products online. For the merchants that are selling products online, parcels are needed to be shipped from the store to their customers. Thus, a delivery service is needed to perform the transshipment. Throughout the delivery process, there is a place whereby all of the products are centralized in a place to perform sorting according to where should the parcel be delivered to. Warehouses and sorting centres are responsible for doing so. At the early stages, all the parcels are sorted manually by humans, which is not efficient and costly in the long run. As the demand of e-commerce rising rapidly, automated sorting centres are being implemented using multiple automated guided vehicles (AGV) or autonomous mobile robots (AMR). Swapping the method from sorting manually to automated able to increase the efficiency on delivering parcels, reducing errors that will be potentially made by humans and able to save lots of labour costs in the long run. These AMRs are robots that are equipped with sensors and navigation technology that able to execute tasks inside a defined environment. AMR able to pick up and transport parcels inside the warehouse. These AMRs are being controlled using a framework to communicate among themselves and some scheduling and pathing algorithms to decides their paths while delivering parcels. In this project, we are further looking into these attributes to produce an efficient automated sorting system.

1.2 Problem Statement

There are a couple of problem that the warehouses are facing while performing sorting manually or implementing automatically which are stated below.

Demands and Customer Expectations

Most of the customers having high demands and they are expecting shorter shipping times, 100% order accuracy, and excellent customer service. And when these expectations aren't met, it can lead to disaster [3]. At a large warehouse, workers may need to spend large amount of time navigating to perform picking and sorting that increases the time needed to ship out customers' orders. Besides that, companies that insists to apply automated sorting needs but wanting to increase efficiency may choose to hire more employees. However, humans may make errors and having some inconsistency while performing tasks comparing to robots. Thus, the order accuracy might not be as high compared to the companies that are applying automated sorting systems. These situations will provide a bad impression to the customers and reduces the profit of the company long term.

Unorganized storage or Insufficient space

Human labour might not be as discipline as robots. While human performing sorting, mistakes might occur such as placing parcels in different places and labelling errors. This may cause the warehouse being unorganized as the parcels are not placed in the correct locations. Furthermore, there are some warehouses with messy and poor layout that might lead to inefficient space utilization. All of these definitely causes some difficulties while performing sorting coherently decreases the overall efficiency. Workers might also find it difficult to proceed with their works with an unorganized, narrow warehouse.

Possibility of Collision between AMRs / AGVs

To solve the issues that exist on manual labour sorting, there are some companies and sorting centres attempting to adopt AMRs or AGVs to automate their warehouses. However, the larger the warehouse, more AMRs or AGVs are needed to maximize the efficiency. One of the challenges adopting automated warehouse is to develop a framework and system to control these automated vehicles. Without a well-design framework, collisions between these vehicles are definitely expected to happen. Collision of these vehicles will slow down the process of escorting the parcels and there might be a possibility of damaging the objects in the parcels. Simple solutions such as stopping the movement of the vehicles able to solve the collision, but it is not the most efficient way as it further lowers the throughput of the vehicles.

Movement Limitations and Warehouse Layout

The speed at which these vehicles or robots move, their degrees of freedom, the ability to drive in reverse, and the turn radius is determined by the manufacturer. Therefore, the ability to deal with reverse driving and turning may be limited to them [4]. Also, they are also having some difficulties travelling on a non-uniformed or unsymmetrical warehouse layout. Without a proper layout and algorithm for the vehicles to move, they might have issues on reaching the destination or even finishing the tasks.

1.3 Motivation

Scalability of e-commerce

Based on [5], The Malaysian eCommerce market grew by 68% year on year in 2021 and it expected to clock around 9.2 billion for the year 2022. Also, the number of online shoppers in

Malaysia grew by 47% year-on-year. From [6], it shows the annual spend in each category on ecommerce in Malaysia. In short, the e-commerce having the potential to scale enormously. Insisting on sorting these goods manually will limit the efficiency on delivering these orders. Thus, adopting an automated sorting centre able to assist on the demand of e-commerce by speeding up the sorting period and shorten the parcel delivery time, simultaneously increasing the traffic on delivering the goods to the customers.

Figure 2 Annual spending on each category [6]

Reducing Labour Costs and Leveraging Human Resources

While human needs some rest after long periods of working, implementing automated sorting system able to operate non-stop without the need of overtime or shift. This method alone able to reduce labour costs and increasing the efficiency in the long term. From [7], in the United States, there are nearly 11 million job vacancies, but only 6.5 million workers are listed as unemployed in 2022. Also, the World Economic Forum states that there will be a global human talent shortage of 85 million workers [8]. In a manually sorted warehouse, there might recruit large number of workers. Performing sorting does not require any skills and can be done easily by robots such as AMRs. Thus, a lot of human resources can be transferred for better used such as on more complicated and requires more critical thinking skills that robots may not be able to compute with. By automating these tasks, humans able to take on jobs that are more challenging and skill-required that robots might not be able to replace.

Resolving Customers' demands and Expectations.

From the problem statement above, it has stated that customers want to receive their orders quickly without mistakes. The sorting centre may face several negative consequences because of making mistake during the sorting processes such as loss of business and reputation damage. Implementing automated sorting centre able to solve both delivery speeds and the correctness of their product. By providing a quality service to the customers, customers will be more likely to continue using the services, gaining better reputation simultaneously increasing the profit of the company.

1.4 Project Scope

The project will develop an automated parcel sorting system and simulates the parcel sorting situation using a simulation software. By achieving that, several elements such as pathing algorithms, framework, task scheduling to the robots and a layout are needed to form a simulation for parcel sorting. This project further looks into pathing algorithms on forming and breaking clusters of robots that moves in the same direction. These elements are developed and improved in order to achieve the most ideal throughput. Besides that, token passing algorithm will be implemented in the robots while performing picking up and delivery. All of these will be implemented in a simulation and should be using as low computational resources as possible to decrease the costs needed as these simulations are able to reflect if the simulations are actually viable to implement in real situation. By using different pathing algorithms and different map layout, the throughput will be compared to find the best combinations of parameters to form the automated parcel sorting center.

1.5 Project Objective

The general objective of this project is to improve the current available sorting centers. One of the goals of this project is to form an optimal cluster while AMRs are moving in the same direction and also find the most suitable time to break the cluster to increase the efficiency of AMRs moving together simultaneously reducing the amount of traffic. Not only that, proposing an algorithm that able to pass tokens among robots that goes together able to solve potential traffics. By using certain pathing algorithms, these AMRs should always find the optimal path while executing tasks or going to the charging stations. From here, the objectives of this project including:

- To find the most optimal way to form or break the clusters when multiple AMRs moves in the same direction while avoiding accumulating traffics.
- To develop a path planning algorithm using token-based approach on a central server and simultaneously avoid collisions and deadlocks.
- To propose an algorithm that enables token passing among robots to achieve better throughput.

1.6 Contribution

This project attempts to find a new way to develop an improved path-finding algorithm for this framework by implementing and modifying other papers strategies. Besides that, this project will also find better optimized way to deal with collisions and deadlocks while the robots are travelling. Also, the projects will propose a new layout design for the robots while considering real-life phenomena such as inertia of the robots and the possibility of robots running out of batteries. By combining different strategies from this project, it should improve the overall throughput and efficiency of an automated sorting system.

By proposing this framework, it can be implemented in real-life sorting centers or warehouses all around the world. By improving the throughput of the sorting system, it will further improve the time needed to deliver these parcels to the customers. Companies that adopting this strategy able to save labour costs on the long run simultaneously reducing any error and mistake that may happened on manual sorting.

Chapter 2 Literature Review

2.1.1 Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification [9]

Introduction

AGVs (Automated Guided Vehicles) in an automated warehouse travel among multiple workstations on a set of predetermined routes. When bidirectional routes introduced for the sake of efficiency, problem of collision arises that may cause deadlock on the route. Thus, the paper presents the types of the collisions that may occur and their respective solutions.

Working Environments

In this paper, they proposed to implement a collision-free route planning for multiple AGVs based on real environment in warehouse. It discusses on different types of collisions and their solutions. Centralized route planning approach is used in the system. From the system, it consists of two layers, upper and lower layer which is coupled with each other. The upper layer which is a server, perform schedules and routing problems whereas the lower layer which consists of several AGVs that is controlled by the server. There is no communication between AGVs. AGVs responsible for pick up, pick down goods and charge.

From the paper, the shelves in the warehouse having a 75cm gap between the ground and the bottom layer of the shelf, which allows the AGVs to pass through the shelf when it is not carrying any loads vice versa. To determine the route of the AGVs, improved Dijkstra's algorithm is implemented to only allows one shortest route for the AGVs. There are 2 different types of tasks performed by the AGVs which is on-load tasks where an AGV picks up goods and transport it to the destination and non-load tasks where an AGV which is not carrying any goods travels from the starting workstation to the destination to pick up the goods located there. Based on that, an AGV that is on-load should avoid route that is occupied by a shelf, but non-loaded AGV do not need to.

Collision Classifications

Since all AGVs are not able to communicate with each other, collision may happen when AGVs are executing their tasks. Four types of potential collisions might happen in the warehouse including:

1. Head-on collision, it happens when 2 AGVs travel the same path from a different location.
2. Cross collision, it happens when 2 AGVs needs to compete for the intersections of the nodes.
3. Node-occupancy collision, it happens when an AGV's route intersects another AGV's goal which blocks the route for the other AGV.
4. Shelf-occupancy collision, it happens when a loaded AGV is being blocked by the shelf that is included in the route of that AGV.

Figure 3 illustrates the collision mentioned.

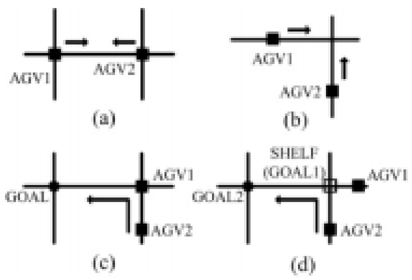


Figure 3 Collision types a) Head-on collision, b) Cross collision, c) Node-occupancy collision, d) Shelf-occupancy collisions. [9]

To resolve potential collisions that may occur, 4 methods are proposed including:

1. Selecting the candidate route, the scheduled of the current route will be changed however the total travel time remains.
2. Waiting for a short period of time, AGV put off starting at the start workstation for a predetermined period until the AGV on the route leaves.
3. Modifying the route, by implementing Dijkstra's algorithm, the route of an AGV will be replanned when the route of the AGV having collisions.
4. Re-dispatching tasks, performing the task by a schedule algorithm.

By testing the 4 proposed methods on the 4 different types of collisions, the paper found that solution (1) and (3) is the best approach for dealing with head-on collisions, node-occupancy collisions and shelf occupancy collision whereas solution (2) is the best approach for dealing with cross collisions.

Strength and Weakness

One of the strengths of the paper is that they clearly detect every possible collision including potential obstacles such as shelf in their system. The layout of the system is fully utilized as non-loaded AGVs are able to pass through the shelf, maximizing the amount of the node in the layout. However, since the paper are focusing on solving all possible collisions, the performance of their throughput decreases, and the time spent increases as some of the solutions requires AGVs to wait and do nothing to solve collision.

2.1.2 Path-planning of Automated Guided Vehicle based on Improved Dijkstra Algorithm [10]

Introduction

In the discussed paper, Automated Guided Vehicles (AGVs) are tasked with moving goods from a specific start point to a destination within a storage setting. The focus is on path-planning to ensure these tasks are completed effectively, with minimal energy usage. The paper mentioned Dijkstra Algorithm and its enhanced variant of the algorithm, particularly tailored for path-planning in the rectangular layout of an Automated Storage and Retrieval System (AS/RS).

Traditional Dijkstra Algorithm

The Traditional Dijkstra Algorithm methodically seeks for the lowest cost path which is the shortest by prioritizing paths of increasing length. It initially finds the shortest possible path from a starting node to adjacent nodes. Then, the end node of this path becomes a new starting node which the algorithm searches for shortest path from this new starting node to its neighbours. This process repeats until all nodes have been visited. A key feature of Dijkstra's Algorithm is that any segment of the shortest path is itself the shortest route from its start to end node. Therefore, the algorithm effectively determines the path that has the lowest cost from the start to the end. However, in AS/RS systems, where workstations are mapped out in rectangular or square layouts, multiple shortest paths may exist, but the algorithm typically identifies only one. The simulation results demonstrating this are shown in Figure 4.

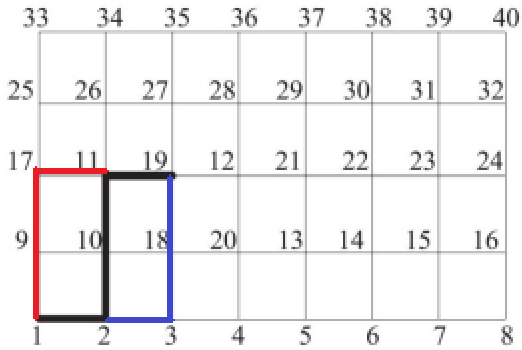


Figure 4 Simulation results of traditional Dijkstra algorithm [10]

From Figure 4, there are 3 paths that is able to reach from node 1 to node 19. However, the algorithm only able to find out one shortest path with 2 turns which is the path of the black line. Hence, the paper intends to improve the existing algorithm to find out all the possible paths.

Improved Dijkstra Algorithm

The traditional Dijkstra Algorithm is enhanced to address the limitation of identifying only a single shortest path in environments, like rectangles, where multiple shortest paths may exist. The original algorithm stores only one intermediate node, limiting it to a single shortest path discovery. The improved version addresses this by keeping all nodes at the same distance from the starting node as potential intermediate nodes.

Additionally, travel time is a crucial factor in path planning. When an AGV traverses a node, it may either continue straight or turn. Straight movement doesn't require slowing down, but turning does. For paths of equal distance, the one requiring fewer turns will be faster. Therefore, alongside identifying the shortest path, the improved algorithm also counts the number of turns, optimizing for both the shortest distance and travel time. The improved Dijkstra Algorithm thus considers the least number of turns to identify the path that minimizes both distance and travel time. By running through simulations, it demonstrates that the improved Dijkstra algorithm able to find out all the shortest path and also find out the optimal paths for the AGVs.

Strength and Weaknesses

The paper solves the problem of traditional Dijkstra Algorithm as the algorithm only able to store one intermediate node hence only one shortest path can be find. This widely increase the routing choices for AGV in an AS/RS system. The improved Dijkstra Algorithm also suitable for bidirectional path-finding. However, the paper still does not solve the greedy nature of the algorithm, as the algorithm only chooses shortest available path, it may neglate some longer path that eventually leads to overall shorter path or shorter travelling time. Besides that, the algorithm only suitable for small to medium layouts, as the larger the layout may consume more memory and time.

2.1.3 Dynamic Resource Reservation Based Collision and Deadlock Prevention for Multi-AGVs [11]

Introduction

Automated Guided Vehicles (AGVs) are increasingly vital in warehouse and industrial settings. Their ability to interact smoothly is critical, necessitating efficient online supervisory control to manage collisions and deadlocks. This paper introduces enhancements to collision and deadlock resolution efficiency in dynamic environments, proposing a Dynamic Resource Reservation (DRR) approach to prevent collisions.

Enviroment Description

Figure 5 shows that the layout proposed having uniformed sized square blockes. Yellow blocks represent AGV's parcking places and green blocks indicates the navigateable paths for AGVs. In the defined areas, AGSv can pause or turn within this specific zones and position in each square. AGVs are restricted to move to the next adjacent blocks when decelerates.

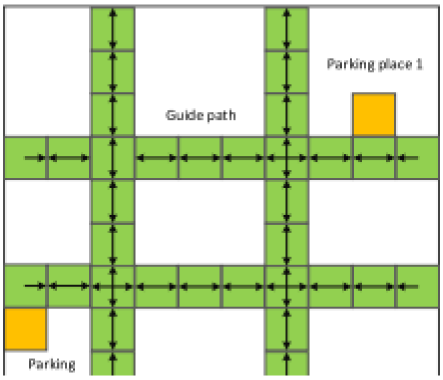


Figure 5 Layout of the System [11]

DRR-Based Collision and Deadlock Prevention

The paper introduces a method for preventing collisions and deadlocks in transport systems using a Dynamic Resource Reservation (DRR) approach. This involves calculating shared resources, detailing how AGVs reserve these points, and implementing control policies through both central and local controllers.

AGVs often face collisions due to limited space. The paper outlines that each AGV can only hold one resource point at a time. Five key definitions for shared resource calculation as below:

1. AGVs in the system can be idle, moving and waiting states
2. Resource points can be idle or occupied by AGVs inside the system
3. AGV's travel path is described with (n_x , n_{x+1}) where n_x is the current point whereas n_{x+1} is indicating the next target point.
4. The remaining transport route of AGV indicates the sequence of points that remains visied by the AGV before task completion.
5. For an active AGV, it consists an sequence of shared resource points with all other AGVs.

Collisions, Deadlocks and Conditions

Figure 6 shows 2 types of deadlock which is heading-on deadlock and loop deadlock. Heading-on deadlock occurs when 2 AGVs travelling at adjacent point but the directions of those AGVs are opposite and Loop deadlock occurs when the travelling points of the related AGVS forms a closed loop

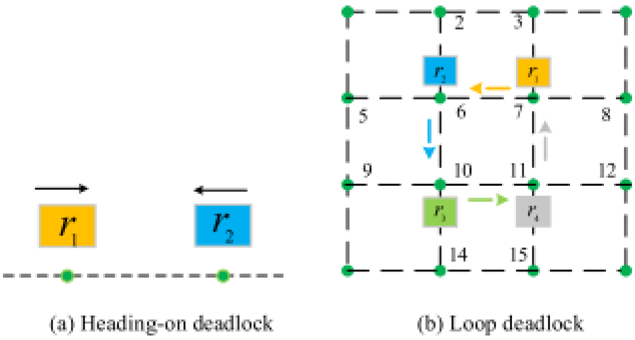


Figure 6 Types of Deadlocks [11]

Besides that, Figure 7 shows that there are two types of collision in the system which is pursue collision and cross collision. Pursue collision occurs where an AGV hits another AGV on the rear and cross collision occur when 2 AGVs wants to have access within same point together.

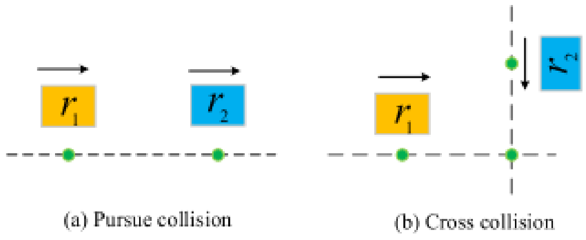


Figure 7 Types of Collisions [11]

To prevent these issue that is mentioned, the paper adjusted to realizing the changing motion states of AGVs according to the state of the resource points. 2 conditions are given to determine whether AGV able to go to the next points.

1. The next points is not the part of any other AGVs path.
2. The next points of an AGV belongs to the guide path of other AGVs but no shared points are reserved by other AGVs.

An AGV are able to reserve their next point if either condition that is stated are met. Otherwise, they need to stop.

From the simulation they compares 2 methods which is DRR and COR, the paper shows results that the total travelling time of DRR is 11% to 28% less than COR with different number of AGVs while maintaining nearly simillar travel distance

Strength and Weakness

The strength of this paper is the paper considers real-life phenomena that may occur real life instead of using many assumption. One of the example that is performed in the paper is deceleration of an AGV that is included as a consideration while detecing resource reservations. Also, the paper compares DRR and COR in detail that strongly shows that DRR is better in several aspects. However, the paper does not forseen unexpected event such as the AGV might be stucked or out of battery causes potential collisions or deadlock. Besides that, the layouts and the conditions might not able to fit in larger amount of AGVs as it may causes congestions between AGVs.

2.1.4 AGV Path Planning based on Improved A-star Algorithm [12]

Introduction

Optimizing AGV's path is one of the most effective way to improve efficiency on AGVs. The paper proposed to improve a path planning algorithm which is A-star algorithm using the characteristics of jump point search to improve the node search mode and the search speed.

A-star algorithm problems and improvements

The A-star algorithm, which relies on actual and heuristic values of nodes, often encounters scenarios where multiple nodes have the smallest values, leading to uncertainty in finding the optimal path. A notable limitation of the A-star algorithm is its inability to choose the optimal path with the fewest inflection points among symmetric paths of equal length. To address this, the paper introduces enhancements by integrating angle cost estimation and node culling based on omnidirectional movement, giving priority to paths with fewer inflection points.

The A-star algorithm is further refined by incorporating jump point search techniques. This approach identifies points that enable 'jumps' directly from the start to the end point, bypassing certain nodes. For example, as shown in Figure 8, Point S can directly reach Point G, skipping nodes n2 and n4. This method doesn't expand intermediate nodes between jump points (S and G), reducing computational and storage requirements. Jump point search is thus utilized to optimize the processing of nodes in the A-star algorithm's open list.

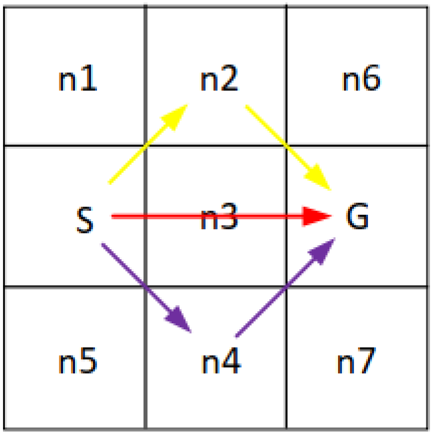


Figure 8 Jump point Searching [12]

Simulation and Comparison

Two different environments are proposed for this simulation, blank grid map and set grid map. For the result for blank grid map, both standard and improved A-star algorithm able to find for symmetric paths of same length but A-star algorithm have more inflection points compare to improved A-star algorithm.

As for the result for set grid map, by performing multiple test in the same start and destination point, A-star algorithm having different path trajectory and the number of inflection points for each path are not the same. This proves that A-star algorithm struggles to come out with the most optimal path. On the other hand, the improved A-star algorithm has stable search results thus able to search for the optimal path. Thus, the paper proves that their improved A-star algorithm offers quicker pathfinding compared to A-star algorithms and able to find a consistent optimal path.

Strength and Weaknesses

The paper significantly improves the existing A-star algorithm on the speed and consistency. It also decreases the amount of calculation needed while finding the optimal path. Also, the algorithm able to reduce the number of inflection points while finding an optimal path, which able to decrease the travelling time of AGVs as the amount of rotation perform by the AGV increases the time. However, the improved algorithm are only optimized on grid layouts and may not be as versatile in other type of layouts.

2.1.5 Simulation-based Multiple Automated Guided Vehicles Considering Charging and Collision-free Requirements in Automatic Warehouse. [13]

Introduction

Automatic warehouse able to provide multiple advantages on productivity, improved quality consistency and lower down the cost and dependence on human labour, thus lowering costs and minimizing errors. The operations in the automatic warehouse were simulated find out all the scenarios that might happened. The paper considers the number of AGV and also the optimal number of workstations for the warehouse and provide a suggested layout and settings.

Problem Formulation

The system includes 3 major system includes AGV control system, vehicle dispatching and tasking assignment problem on the workstation.

The paper introduces an Integrated AGV system which combines both hardware and software, primarily managed through a computer program. This system is designed to manage multiple AGVs and integrate them into the warehouse's internal operations. A key aspect of the system is frequent unit load servicing, necessitating efficient vehicle dispatching. This involves selecting an AGV for material pickup or delivery, with dispatching issues categorized into two types and addressed through heuristic rules.

Another significant aspect discussed is sensor, zone, and deadlock control. AGVs are equipped with sensors to detect other vehicles, enhancing safety and efficiency. The paper discusses implementing zone control, wherein guide paths are segmented into distinct zones, with each zone accommodating only one vehicle at a time, as demonstrated in Figure 9.

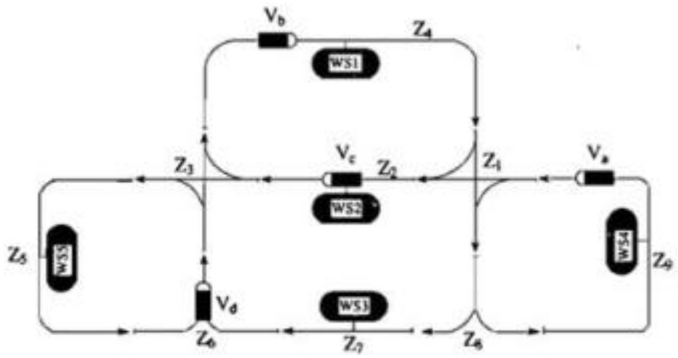


Figure 9 Scenario of zone control [13]

By dividing the guide paths into zones, it reduces the chance of collision but does not eliminate it. Thus, the paper resolves the deadlock problem by adopting Binary Semaphore and Counting Semaphore in intersections and bidirectional paths.

Simulation Environment and Results Summarization

The paper proposes a simulation environment that is shown at Figure 10. which firstly includes control points for the AGVS to follow the paths. The simulation also includes corridors, workstations and charging stations. There are 2 types of paths which is the main path and the spurs path which main paths are unidirectional and spurs paths are bidirectional.

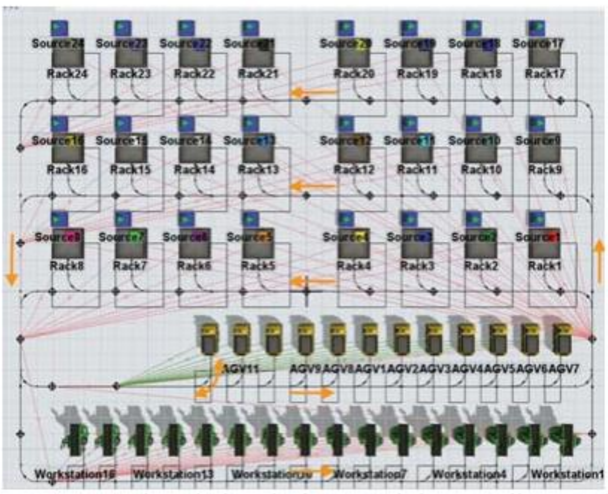


Figure 10 Environment overview [13]

For the simulated results, the paper discusses the optimized models by comparisons between each other in several measurements including total maximum output, maximum output per workstation and maximum output per AGV. The output results are measured and compared by the number of workstations. From all the results shown, there is a trend that the first few workstations added contributes significant to the output until a specific amount reached.

Strengths and Weaknesses

The paper clearly describes the flow of the AVG in an Automatic Warehouse. The paper considers real life challenges such as AVGs need to be charged thus setting up a charging station. However, the layout provided by the paper may not be suitable for our research as the layout is not a grid layout. The layout they provided could causes heavy traffic jams while an AGV want to leave the destination.

2.1.6 An Overview on Swarm Robotics: Swarm Intelligence Applied to Multi-robotics. [14]

Introduction

Swarm intelligence is a new field to build a fully distributed de-centralized system with the functionality of the interaction of individual agents able to interact with each other and its envorment. One of the domain, swarm robotics (SR) has been issued to study how a large group of robots able to communicate and coordinate with each other using rules. The paper proposed an overview of SR including some functionalites and potential applications.

Swarm Intelligence

Swarm intelligence, which is inspired by the behaviour exhibited by natural grouping systems in the natural world, consists of a population of relatively simple agents which interacts only locally with themselves and with their environment without having global knowledge. These theories are widely applied to solve analogous engineering problems.

Multi-Robot Systems (MRS)

Multi-Robot systems (MRS) exists to overcome the lack of information processing capability and many aspects of a single robot that are not able do deal with some certain tasks. Thus, MRS introduces cooperation and collaboration between groups of robots. MRS having the advantage of simplicity in design and increasing in flexibility, capabilites and fault tolerance. However, challenges such as decentralization in control and self-organization introduces a new sub-domain wich is swarm robotics (SR)

Swarm Robotics (SR)

Swarm Robotics (SR) is a peculiar sub-area of collective robotics that applies the techniques of swarm intelligence. The idea of SR is to build some relatively many small and low-cost robots that able to acoomplish the same assigned task as a single complex robots or a small group of complex robots. Comparison between MRS and SR are shown on Figure 11.

	Swarm robotics	MRS
Population Size	Variation in great range	Small
Control	Decentralized and autonomous	Centralized/ remote
Homogeneity	Homogeneous	heterogeneous
Flexibility	High	Low
Scalability	High	Low
Environment	Unknown	Known/unknown
Motion	Yes	Yes

Figure 11 Comparison between MRS and SR [14]

Potential Applications of Swarm Robotics

There are several work field has been issued to explain the benefits from the properties of swarm robotics systems in several potential application domains.

1. Task covering large area
   - These tasks are most suitable for robots swarms as they are able to act autonomously without any external coordination.
2. Tasks dangerous to robots
   - Economy acceptable to use swarm robotics with cheaper individuals rather than expensive robots
3. Tasks require scaling population and redundancy
   - Best to apply in situations which is difficult to estimate resource needed in advance as the robots can be removed or added in them hence making it scalable.

Strength and Weaknesses

The paper discuss the overview of swarm robotics, which is one of the method for AGVs to work with each other by forming a small cluster to increase the overall throughput of completing the tasks. However, further evaluation and reseach is needed to apply swarm robotics into AGVs to avoid potential collisions and deadlock.

2.1.7 Lifelong Multi-Agent Path Finding for Online Pickup and Delivery Tasks [15]

Introduction

Most of the research before are concentrated with multi-agent path finding (MAPF) problem which has been studied in various field. In MAPF, each agent has to move from its current location to the destination without collisions while having the same number of agents and destinations. However, it does not cover real-world problem such as autonomous warehouse where the number of agents may not be the same as the number of destinations. Thus, multi-agent pickup and delivery (MAPD) is further study in this paper, which these agents need to attend a stream of delivery tasks in an environment that is modelled as an undirected graph. The paper introduces two decoupled MAPD algorithms which is Token Passing (TP) and Token Passing with Task Swaps (TPTS).

Token Passing (TP)

Token Passing (TP) operates on the principle where agents sequentially plan their paths. TP assumes agents will remain stationary at their final path location as indicated in the token. The algorithm operates as follows:

1. The system initializes the token with basic paths, positioning agents at their starting locations, and adds new tasks to the task set.
2. Agents at the end of their path request the token once per timestep.
3. The system grants the token to requesting agents.
4. The token-holding agent selects a task from the task set, ensuring no other agent is assigned the same delivery location.
5. If suitable tasks are available:
    a. The agent picks the task with the shortest travel distance, removing it from the task set.
    b. The agent updates its token path to a collision-free, cost-effective route with the same destinations.
6. If no suitable tasks are available:
    a. The agent doesn't take on a new task.
    b. The agent updates its token path, unless already at the delivery location.
    c. If at the delivery location, the agent alters its direction to prevent collisions.
7. The agent returns the token to the system and follows its designated path.

Token Passing with Task Swaps (TPTS)

TPTS operates similarly to TP, but the task set includes all pending tasks, not just those unassigned. TPTS also assumes agents remain at their path's final location. The TPTS algorithm functions as follows:

1. Similar to TP, agents are given tokens and request tasks. Agents with tokens execute the GetTask function; those without or lacking an endpoint are denied.
2. When executing GetTask, agents consider tasks that don't end at their location.
3. If such tasks exist:
   a. If no agent is assigned, the agent takes the task, updates the token path, and proceeds.
   b. If another agent is assigned, that agent is unassigned, and the new agent takes over, updating the token path.
   c. If the new agent reaches the pickup location quicker, it passes the token to the previous agent, prompting them to seek a new task.
4. If no unique-location tasks exist:
   a. The agent doesn't assign itself a task.
   b. If not at an endpoint, the agent seeks a path to one, indicating success or failure based on path availability.
   c. If at an endpoint, the agent updates the token path accordingly, marking success.
5. The agent then returns the token to the system and follows its token path.

Centralized Algorithm

A centralized MAPD algorithm is used to evaluate TP and TPTS effectiveness:

1. The central system assigns endpoints to all agents and solves the MAPF instance for path planning.
2. It considers each agent at a task's pickup location.
3. The central system constructs a set of potential endpoints for free agents.
4. It then assigns each free agent an endpoint.
5. The system employs an optimal MAPF algorithm for planning collision-free paths.

Results

Both TP and TPTS able to handle and solve all the MAPD instances. The MAPD algorithms in services times of the algorithm in increasing order is Central, TPTS and TP. Whereas the MAPF algorithm’s orders is TP, TPTS and Central. Overall, TP can be extended to fully distributed MAPD algorithm and is the best choice for real-time computation whereas TPTS requires limited communication among agents and balance between TP and Central.

Strengths and Weaknesses

The paper discuss token-based algorithms, which is very suitable to apply on the automated vehicle to increase the throughput. Token-based algorithms able to avoid collision without further deploy another algorithm purely for collistion detections. Besides that, Task swaps enhances flexibility that able to change certain sorting requirements or unexpected events such as unexpected obstacles or traffic congestion. Also, Token-based algorithms able to implemented in both centralized and decentralized system thus is much more flexible and robust.

However, Token-based algorithms needs more computational power as it may often requires real-time calculations such as task swapping that may often changes the route of an agent. Besides that, possible delays might happen while performing token passing thus synchronization may be hard to perform between multiple agents.

2.2 Comparisons

Comparisons on Centralized and Distributed system.

Centralized	**Distributed**
Single Point Control	Multiple / Decentralized Control
Limitation on Scalability May be difficult to add in more vehicles	Scalability Adding vehicles without increasing complexity
Single point failure Whole system will be down if the central controller fails	Partial Failure Certain Functionalities may be affected
Simplicity All the decision making are done in one system	Higher Complexity Need to perform coordinations and synchronizaion among different servers / vehicles
Minimal Latency / Delay	Potential Latency / Delay

Table 1 Comparisons on Centralized and Distributed system**.**

Chapter 3 System Design

3.1 Design Specifications

3.1.1 Methodologies and General Work Procedures

Figure 12 illustrates the workflow for developing the simulations. The process begins by establishing basic robot movements and designing the initial grid layout. Subsequently, path planning is integrated, enabling the robots to determine the most efficient travel routes. Once the fundamentals are in place, tasks are allocated to individual robots, followed by simultaneous navigation of multiple robots within the layout. To address potential collisions, directional constraints and a token-based path allocation system are incorporated into both the layout and robot coordination. Fine-tuning is essential to optimize the cluster formation during token passing. Finally, the program undergoes thorough optimization and refinement before the simulations are executed.

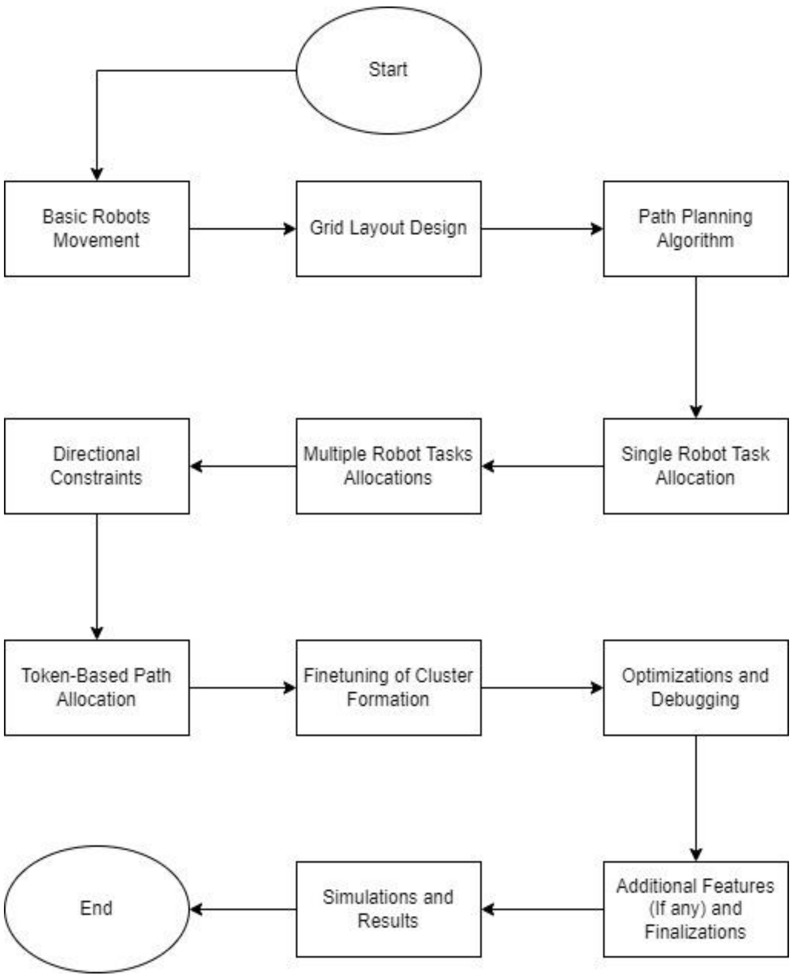


Figure 12 General Work Procedures

3.1.2 System performance definition

The goal of this project is to find the optimal way for the robots to find its path to execute their tasks effectively creating high throughput of finishing the task. For this project, throughput is defined as the number of tasks finished by the robots. A task is finished when a robot picks up a parcel from a pickup point and locates it at a drop-off point. Finishing this cycle counts as finishing a task. Thus, the system performance is determined by the throughput of tasks in a certain amount of time with a certain number of robots in a layout.

To increases the performance of the system, effective path finding of the robots are one of the improvements that can be made to increases the throughput. Thus, robots should be allocated with the shortest possible path simultaneously avoiding collisions and deadlocks.

3.1.3 Verification Plan

The project will be verified through various tests and feasibility studies to ensure that it can be completed without errors. Each procedure will undergo testing to validate assumptions and assess performance under different scenarios, ensuring the system's viability before proceeding to the next stage. After completing the framework, iterative improvements will be made, continuously refining the system based on feedback from the simulations.

Finally, the system's efficiency will be tested against various constraints and attributes, such as the number of robots, layouts, and algorithm parameter tuning to determine which factors affect throughput and to identify the most optimal attributes for the best throughput.

3.2 System Design / Overview

3.2.1 Layout Components and Designs

The robots will be moving on a grid layout design whereas nodes are connected with each other to execute their tasks. The design of the layout will be unidirectional with intersections to solve collisions. The grid of map consists of 4 ways directions (up, down, left, right), intersections, pick-up point, drop-off point and charging point. Directions grids strictly limits the movement direction of the robots whereas intersection are the grids that robots able to makes it own decisions. Pick-up points and drop-off points are the tasks that the robots assigned to pick up an object and deliver it to the drop-off point. Charging points are provided for the robots to charge its battery when it falls over a certain threshold.

3.2.2 System Architecture Design

Token-Based System will be using both centralized and decentralized architecture design for the robots to perform tasks. Centralized architecture is being used for path planning on robots and token allocating for each grid cell. The robots however navigate their path by its own without knowing other's agent path which is a decentralized architecture. The allocator of the system haves the information of the layout on each grid and the current location of all the robots.

3.2.3 Possible Deadlock and Collisions

A collision in this context, refers to an event when 2 or more robots occupies the same grid cell at the same time. Deadlock on the other hand refers to a situation where these robots are waiting each other to make an action, causing them to be in idle state and does nothing. When the robots are travelling, there are possibility that 2 or more robots needed access to the same grid at the same time. Thus, collisions happens when a robot collide with another robot from other directions. Unidirectional lanes are implemented to solve collisions however collisions mostly happen when robots attempt to access an intersection grid cell. At the intersection grids, there are a lot of potential robot movements that causes collisions. Furthermore, deadlocks arise in a system where the resources of the layouts are limited causing robots unable to move. Figure 13 illustrates the possibilities of deadlocks and collisions that might happen from multiple directions when a robot travels.

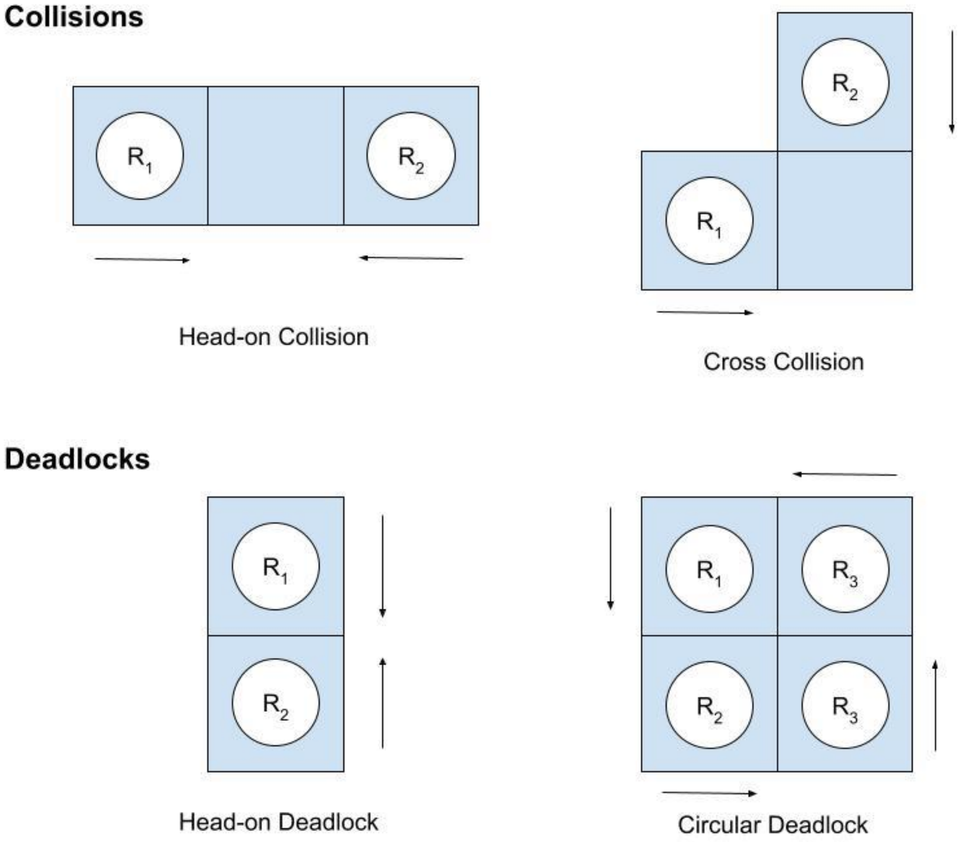


Figure 13 Collisions and Deadlocks

3.2.4 Path Finding and Token Based-Approach

To find the nearest path for the robots to travel, a path finding algorithm named Dijkstra algorithm is being used. From [10], Dijkstra Algorithm seeks for the shortest path in the order of increasing path length. It will find the shortest path from source node to its adjacent nodes first. However, there might be a chance that robots may be using the same node at the same time causing collisions in the intersections. Thus, Token Based-Approach is being introduced to solve the potential collisions and deadlocks. Figure 14 shows the concept of Token-Based Approach.

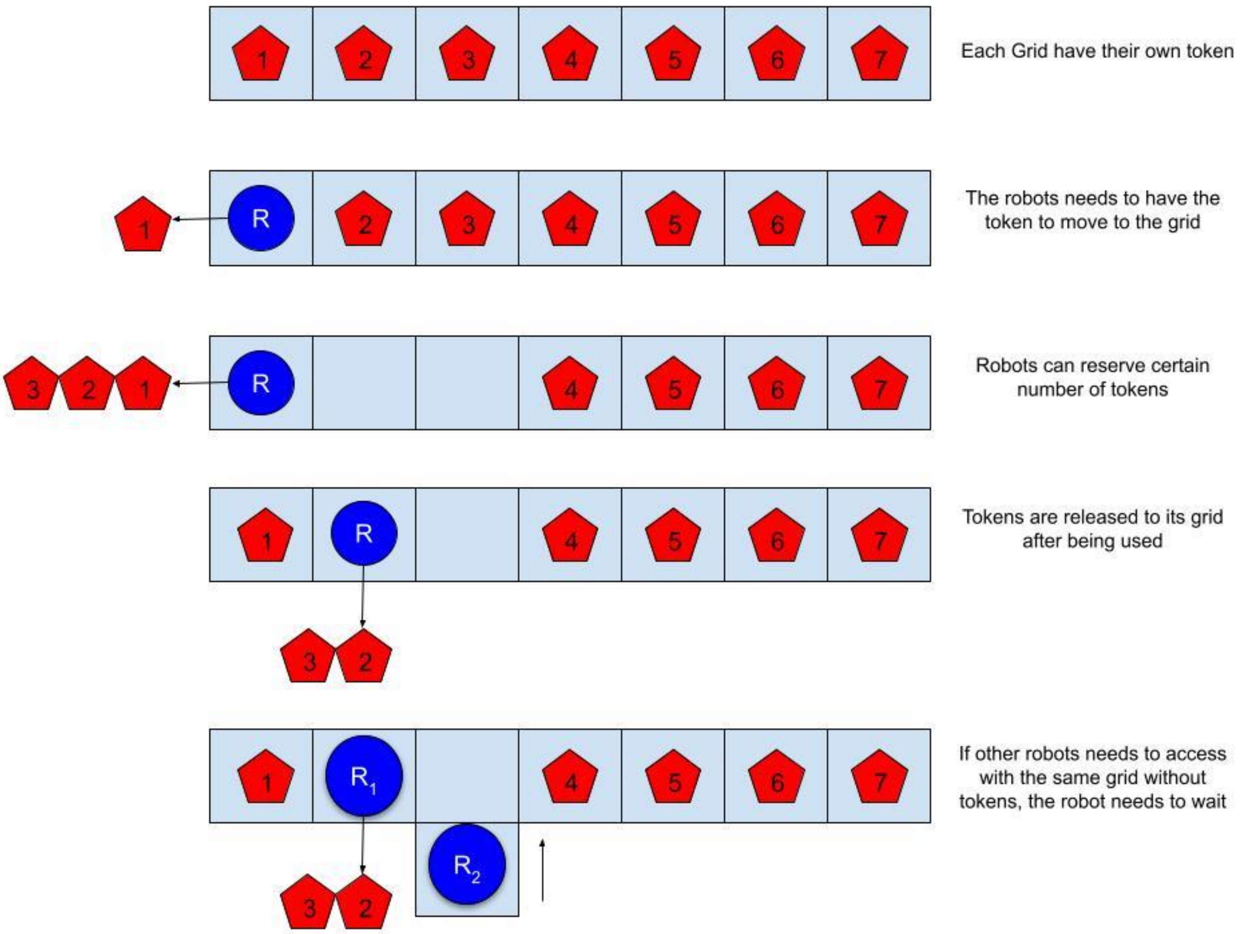


Figure 14 Token-Based Approach

On each individual grid on the grid layout, it consists of a token. When a robot needs to go through a path for each grid, they need to first request the token in order to get the access of using the node. The robots can get multiple of tokens to reserve more grids by providing a quota. The robots only can use the node when they have the token of the grids. Robots that fail to acquire a token have to wait or search for an alternative path. After passing through the grid, the robot will release the token instantly back to the server or passes to another robot which is right behind the robot. The process continues until the robot reaches its destination. Implementing token-based approaches solves 2 of the collisions that is mentioned in Chapter 3.2.3

3.2.5 Robot Movements on Grid Layouts

Robots will move grid by grid on the layout. Hence, these robots will be moving at the same speed. Robots also obeys all the unidirectional grid cell except entering their destination. Assume that the intersection grid is restricted by its connected unidirectional grid, all possible robot movements will be shown in Figure 15. Thus, it shows that robots from all directions having a chance to uses all the intersection grid thus causing potential collisions and deadlocks. When all the robots requested the next occupied grid and moves at the same amount of speed, the circular deadlock which is presented at Chapter 3.2.3 can be solved in the intersection grids.

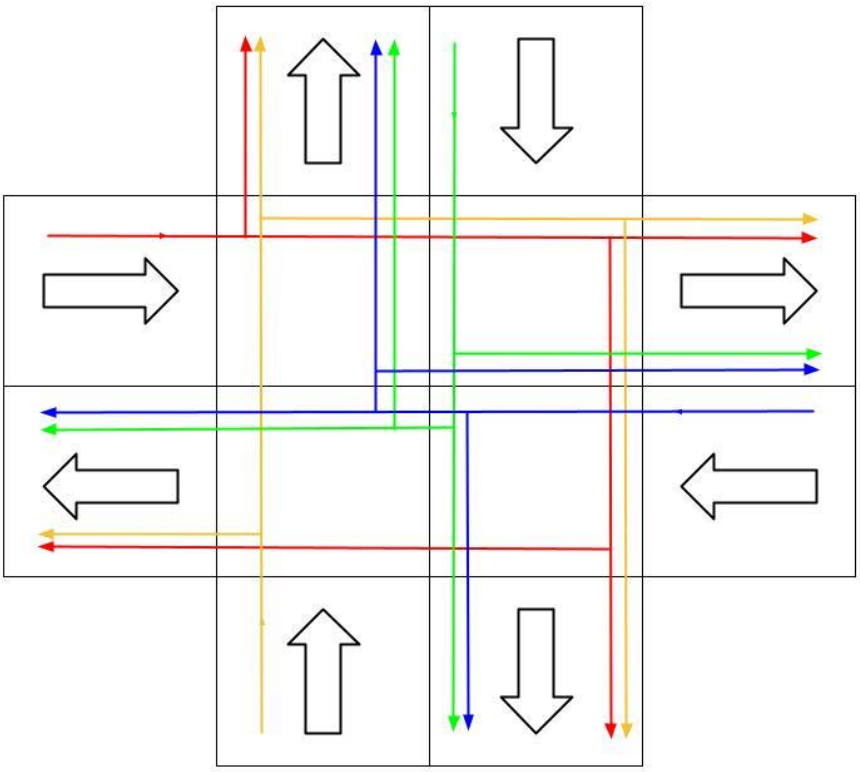


Figure 15 Possible Robot Movements

3.2.6 Robot's Process flow

Figure 16 shows the flow procedure of the robots. Initially, the robots will be randomly located onto a traversable node in the grid layout, The robots will constantly check if their battery is lower than the threshold which is 20%. If the robot having battery lower than the threshold, it will find a path to a charging point. The robots will firstly find if there is an available charging point. If there is a charging point the robots will proceed with charging. Else, it will be performing sorting tasks. The destination of the pickup point and drop-off point are randomly selected. The robot will continue to perform sorting tasks once it is charged more than 90%. If the robot battery health is not lesser than 20%, it will directly perform sorting tasks. Robots will firstly find the path to the pickup point after being allocated with a task. The robots will first request for grid token to move. If grid token is available, robots are permitted to move forward else need to wait for the token and stay idle. Repeating the steps, the robot will travel to the pickup point that is allocated by the server. After reaching the pickup point, the robot will finish the drop-off task regardless of the battery health. The robots only can proceed to request for charging after finishing their tasks. If the battery health of the robots still remains more than 20%, it will proceed to the next sorting tasks.

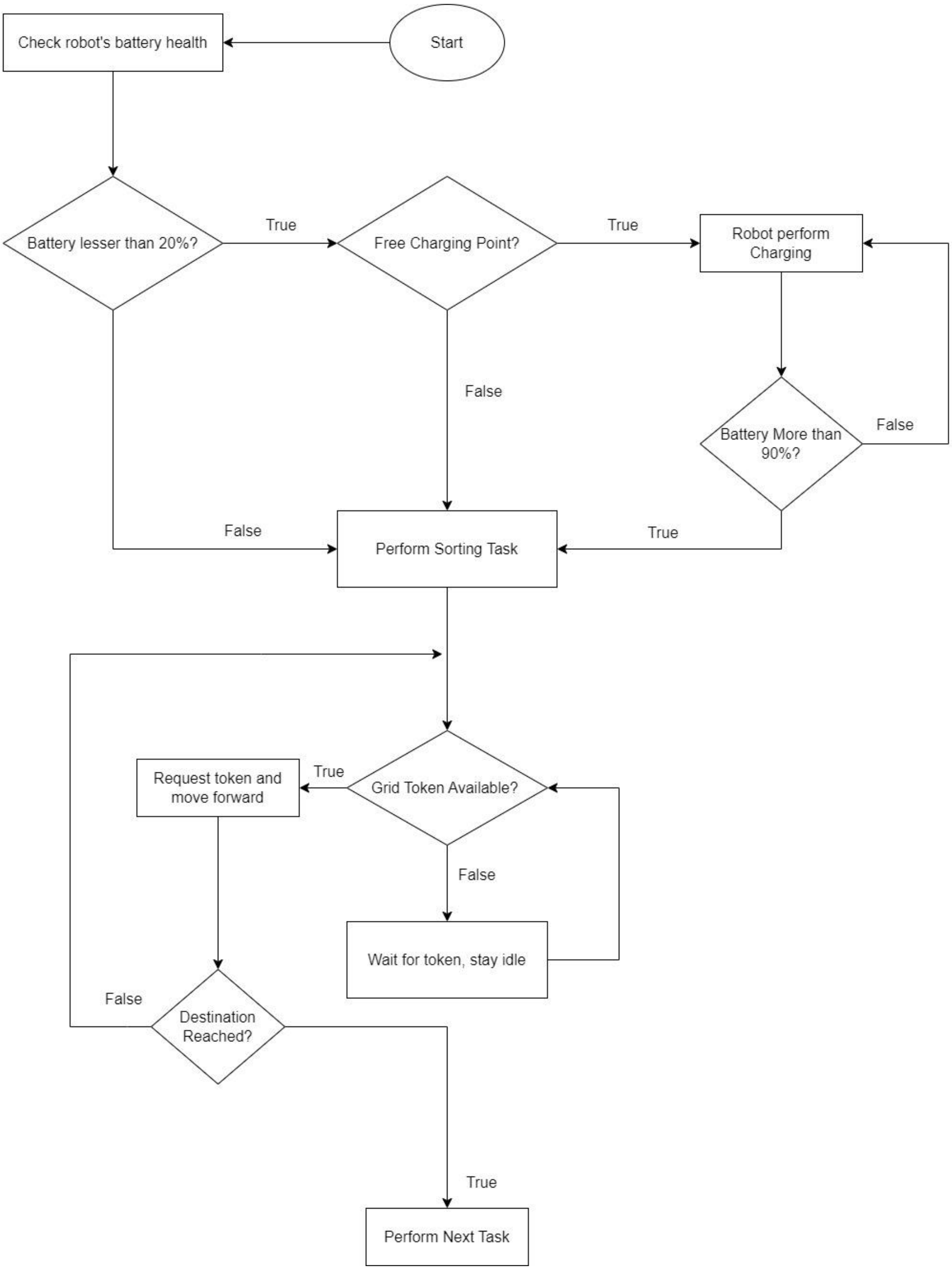


Figure 16 Process Flow of The Robots

3.3 Token-Passing Approach / Design

3.3.1 Highly Contested Intersection Grids

One of the challenges with the token-based approach is the allocation of tokens in a highly contested grid in the intersection. In an intersection grid, robots from both ways are waiting their turn to get the token in order to move to the next grid. However, passing the token to the same direction might causes starvations on the other side but passing the token towards the other side may decrease overall throughput as it breaks the cluster that is moving together. The amount of cluster should be form and when should the cluster be broken is one of the challenges and it is a factor to find out the most efficient cluster formation.

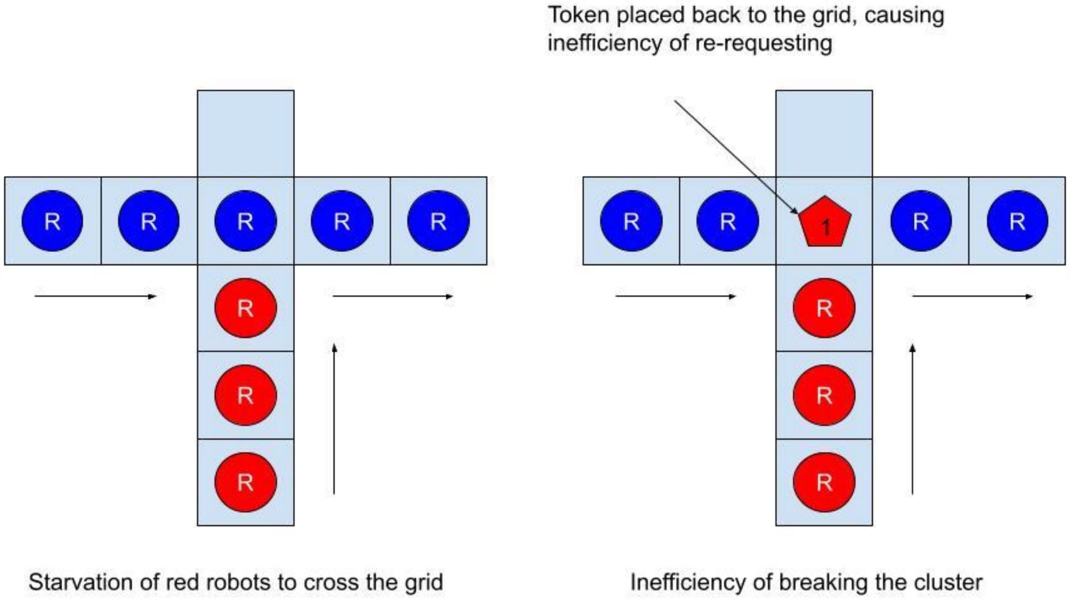


Figure 17 Token-Based Approach Issue

3.3.2 Potential Deadlocks Using Token-Based Approach

In 3.2.3 and 3.2.5, we have discussed the ways in which we can solve the circular deadlock issues which the robots should simultaneously request the same grid cell and moves at the same time to prevent deadlocks. However, using token-based approaches, the robots might break the clusters for the other site of the robot clusters to move which is stated at 3.3.1. Thus, when circular deadlock happens, instead of requesting the next grid cell, the robots will return back the token back to the allocator, which causes that these robots may not request the next grid cell at the same time since returning the token takes time, causing potential deadlock while 4 robots want to move to other's robot locations at the same time. Figure 18 shows the robot requesting tokens when there is a robot in front. From the diagram, the robot not able to request tokens at the same time causes the robot in front will move forward first. In a circular situation, the robots not able to move forward since all of the grids are being occupied yet not able to request to the next grid cell.

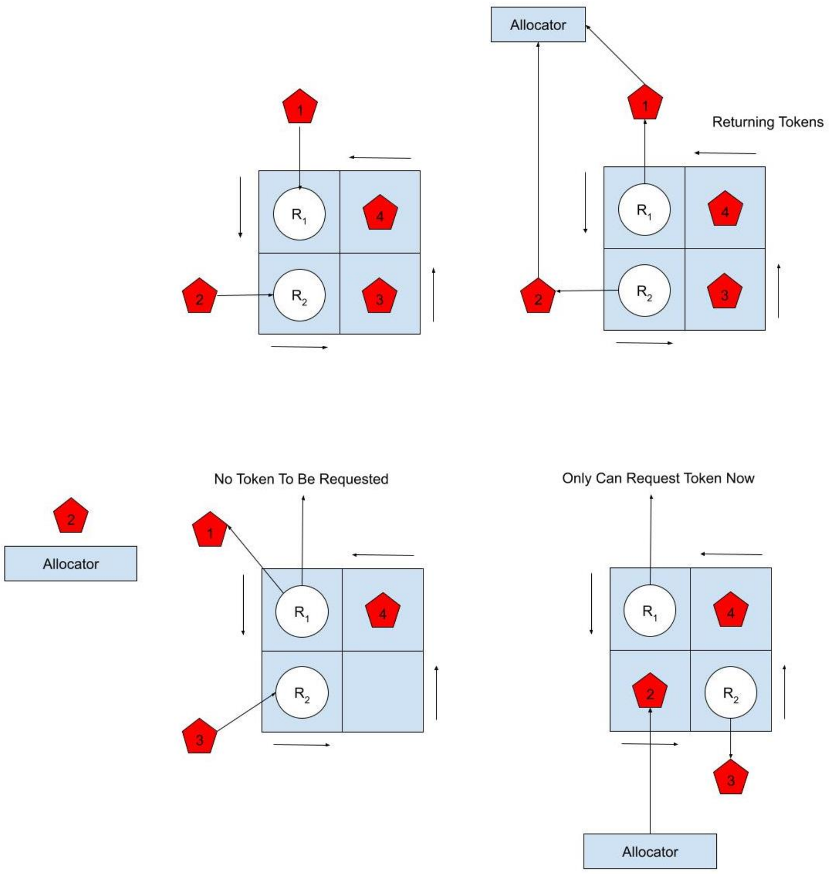


Figure 18 Token Requesting

3.3.3 Token Passing

To approach the issues in 3.3.1 and 3.3.2, token passing approach has been introduced to address the challenges of deadlocks and starvation by utilizing a token-based system. Token passing approaches identifies robots that are directly behind another robot intending to move into the grid cell that the robot is currently standing. Besides that, the functions also prioritize those robots that are currently located on the intersection grids. This approach able to fix the issue that the robots will not pass back the token to the allocator instead passing the token to the robots behind if there is any. Figure 19 shows the flow of token passing approach. From Figure 19, whenever robot wants to request token where the token is acquired by other robot, in other terms, occupied by other robots, it will check whether the robot is the only robot that needs the token. If there are more than 1 robots requesting for the token, it will prioritize the robots that are currently occupying intersection grid and pass the token to the robot first. Prioritize robots on the intersection grid allows circular token passing and block other robots to access the intersection grid where they came from unidirectional grid.

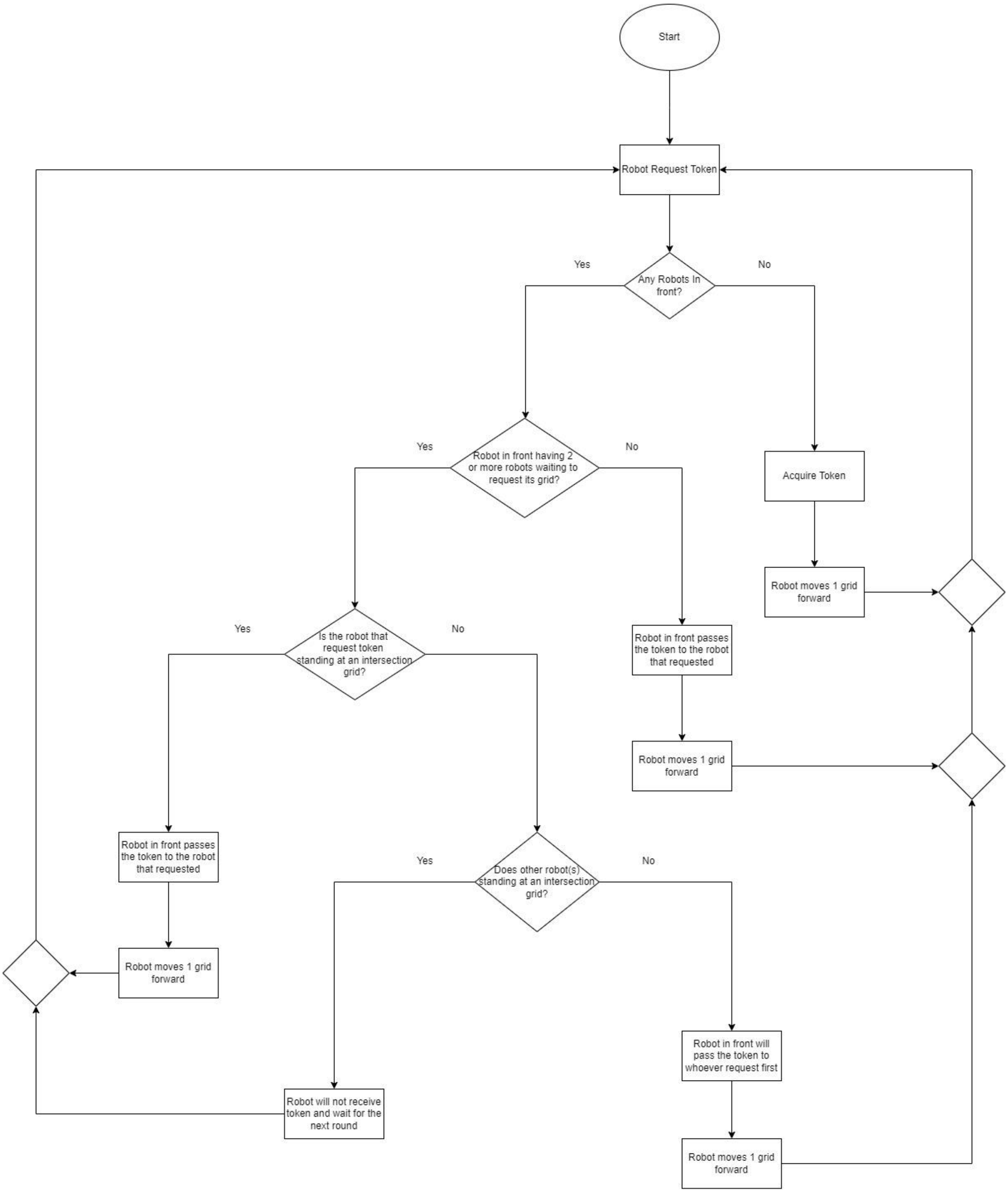


Figure 19 Token Passing Flow Chart

Chapter 4 Experiment and Simulations

4.1 Assumptions on the Simulations

4.1.1 Assumptions on the Robots

The assumption for the robots is as follows:

- The robots able to move for 1 grid per second
- The robots accelerate and decelerates at the maximum speed (1 grid per second).
- The robots can acquire at most 3 tokens for its next 3 grid locations.
- The battery capacity of the robots starts randomly between 10% to 100%
- The battery will drop between 1% to 3% upon reaching pickup and drop-off point.
- The battery will be charging 5% per second.

4.1.2 Assumption on the Grid Layout

The assumption for the grid layout is as follows:

- Size of the grid layout will be 76x76 including boundary, 74x74 without boundary.
- Robots will be randomly placed at unidirectional grid or intersection grid all over the layout when the simulation starts.
- The directionality of intersection grids is determined by the unidirectional grids adjacent to them.
- Dropoff and pickup point can be the destination of multiple robots with a limited amount, whereas charging point can only be one robot's destination.

4.2 Tools to Use

The framework and path-finding algorithms will be written in Python Language using PyCharm. Python language is chosen for several reasons. Firstly, Python consist of rich library system that is able to simply some complex tasks. Python also known as doing scientific and numeric computing which is a wise choice while involving algorithmic programming and data analysis. Besides that, Python can be easily integrated with other languages and tools if needed which can be useful that require a combination of other languages or tools. A laptop is used to develop the all the framework, algorithms, and simulations. Table 2 shows the specifications of the laptop.

Description	Specifications
Model	Acer Swift 5
Processor	Intel Core i5-1035G1 CPU @ 1.00GHz
Memory	8GB DDR4 RAM
Operating System	Windows 10
Graphic Card	Intel Graphics
Storage	256GB SSD + 256GB HDD

Table 2 Laptop Specification

4.3 Settings of the Map Layout

Figure 20 shows the layout of the map. The map is referred from the project that has been initially done by previous project [16]. The arrows represent the unidirectional grids, the green box represents pickup point, yellow box represents charging station and green box represents drop-off point. Blue dots are the robots that executes their tasks by picking up items and dropping them off.

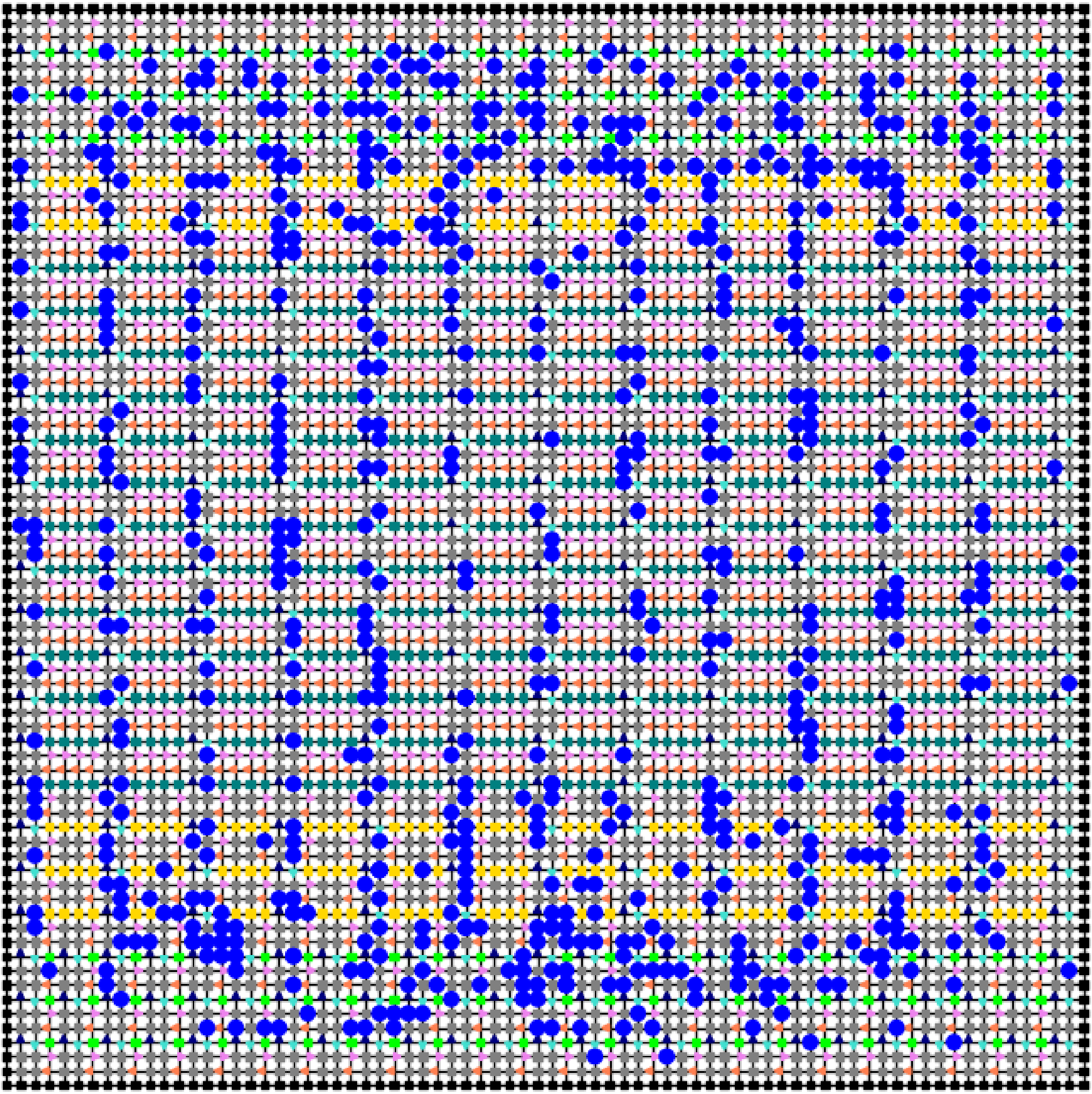


Figure 20 Map Layout

4.4 Implementation Issues and Challenges

4.4.1 Greedy Allocator

A significant challenge in the token-based simulation is the implementation of a greedy allocator, which is designed to assign robots to pickup and drop-off locations based on immediate availability without considering the overall system state. This approach, while efficient by increasing the throughput rate, assumes that robots can be assigned to the same locations simultaneously. Although the large size of the map (74x74) significantly reduces the likelihood of congestion and related deadlocks, the risk increases with the number of robots in operation. Specifically, deadlocks may occur if robots destined for the same location arrive in succession, particularly one directly behind another. Figure 21 illustrates the potential scenario. Assuming the robots are heading towards the same destination.

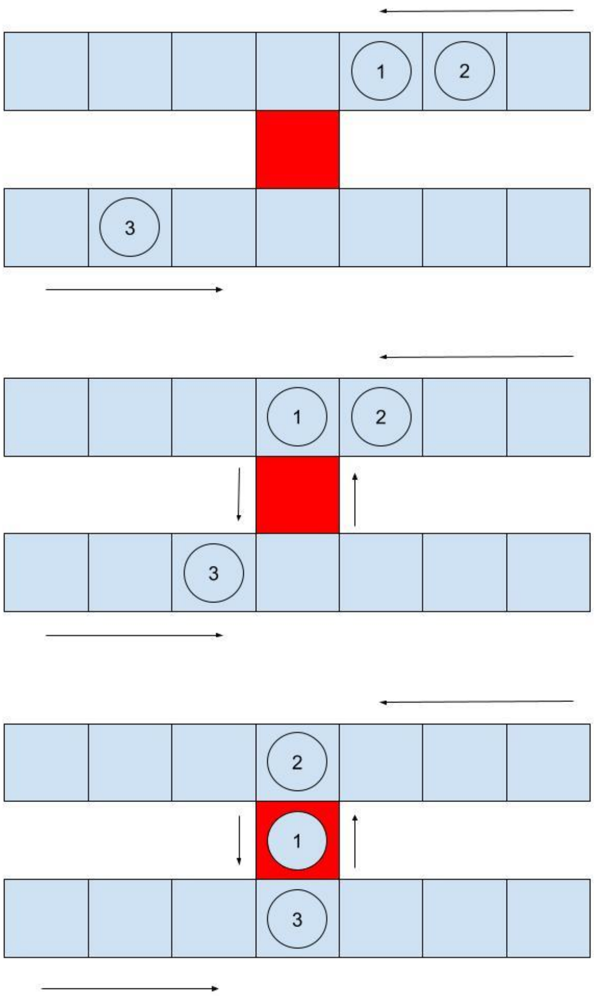


Figure 21 Greedy Allocator Deadlock

When there are 2 or more robots assigned into the same destination and these robots reaches the destination back-to-back, the exit will be sealed off which the robot unable to move out from the destination whereas other robots blindly wait for the grid to be free. This issue, though rare given the scale of the environment, underscores a potential flaw in the greedy allocation strategy that could lead to deadlock. Thus, implementation such as limiting the amount of robot having the same destination or any alternatives should be considered to prevent deadlock.

4.4.2 Starvations on Unidirectional Grids

In our robot navigation simulation, an observed issue which is starvation at unidirectional grids, which becomes increasingly pronounced as the number of active robots escalates. Given the system's prioritization of robots at intersections, robots positioned on unidirectional lanes frequently experience significant delays, forming unexpected long queues. This issue stems from the strategic emphasis on intersection grid traffic, which is critical for managing flow at key junctures, slowly leads to congestion and inefficiencies in less prioritized areas. As the robot count increases, the likelihood of these bottlenecks intensifies, complicating the task of breaking up clusters of robots effectively.

Addressing this challenge requires advanced approach to traffic management that extends beyond simple prioritization of intersections. Implementing adaptive traffic control algorithms that can dynamically assess and respond to congestion levels across the grid may offer a solution. Additionally, exploring strategies for optimally timing the breaking of robot clusters could help balance the flow between intersections and unidirectional lanes. Such changes are essential for enhancing system efficiency and reducing the wait times that lead to starvation in specific grid areas.

4.4.3 Layout Designs.

In our robot navigation simulation, the layout design follows the template used in [16] but with notable challenges affecting system performance. Primarily, the layout does not ensure a balanced distribution of pickup and drop-off points, which has been identified as a potential cause of reduced throughput. From the layout of [16], there is 672 drop-off points, 144 pickup points and 192 charging stations. This imbalance means that certain areas of the grid become hotspots for activity especially in the pickup points, while others are underutilized, leading to inefficiencies in robot routing and task allocation. Additionally, the placement of charging stations between pickup and drop-off locations further exacerbates the issue by increasing the travel distance required for task completion. These layout constraints not only extend the operational time for each task but also contribute to energy inefficiencies and increased waiting times.

To address these concerns and maximize throughput, a redesign of the layout is recommended. This new design should strategically distribute pickup and drop-off points to ensure more uniform robot traffic and utilization across the grid. Including conditions that optimize the placement of charging stations to minimize their impact on travel distances between key operational points is also crucial. By reevaluating the current layout and implementing a more balanced and strategically thought-out design, the system can achieve higher efficiency, reduced travel times, and ultimately, an increase in throughput.

4.5 Concluding Remarks

Despite the challenges that is mentioned including issues with the greedy allocator, starvations at intersection grids and suboptimal layout designs, the project still presents a robust platform for advancing automated sorting centre.

The use of a greedy allocator, despite its potential for causing deadlocks, offers simplicity and speed in decision-making which is invaluable for testing under varied scenarios without the overhead of more complex algorithms. Without having the assumption of multiple robots can assigned to the sane direction, there will be robots that does not have a location, potentially causes big traffic as the robot is not moving.

The issue of starvation at intersections, while highlighting the limitations of current traffic prioritization schemes, provides unique insights into dynamic traffic management and the effects of strategic prioritization among intersection grids. Without prioritize the intersection, the intersection might have heavier traffic that likely causing potential deadlock.

Meanwhile, the existing layout design, despite its imbalance, has enabled us to identify key factors that influence throughput and efficiency, offering clear pathways for iterative improvements. The layout has more potential upon changes based on the robot amount and assumptions of the layout.

Ultimately, these challenges are instrumental in driving forward the development of more of a intelligent autonomous sorting centre. They help in pinpointing areas where adaptive strategies and innovations are needed, thereby contributing to achieve the goals of enhancing autonomous sorting centre. Through continuous improvement and by addressing these identified issues, the project is well-positioned to develop solutions that could eventually be applied in real-world settings, significantly impacting how automated systems operate in complex environments.

Chapter 5 System Evaluation and Discussion

5.1 Performance Measurement and Definition

To evaluate the performance of our robot navigation system, throughput per hour serves as the unit for our testing. It quantifies the efficiency of the system by measuring the number of tasks completed within a specified timeframe, which is bringing an object from the pickup point to drop-off point. Besides that, throughput is assessed across configurations utilizing different numbers of robots. By increasing the number of robots incrementally, we can observe the relationship between system capacity and operational efficiency, identifying the optimal balance where throughput maximizes without compromising the system's responsiveness and reliability. This method allows us to determine the thresholds at which additional robots either contribute to a linear increase in productivity or lead to diminishing returns due to congestion and increased coordination overhead.

5.2 Testing Setup and Result

5.2.1 Testing Setup

Simulations will be performed under different number of robots to find the peak throughput.
To evaluate the efficiency of our robot navigation system, simulations are conducted under varying robot numbers to identify the configuration that achieves peak throughput. These tests involve systematically increasing the number of robots operating within the simulation to determine the maximum effective operational capacity. Furthermore, a comparative analysis will be undertaken against the project [16] to find the relative capacity of the system. This comparison aims to benchmark the maximum number of robots that can be effectively deployed in our simulation against [16] systems, providing a direct measure of our system’s scalability and performance. The results of these tests will not only highlight the optimal robot density for maximizing throughput but also determine the relationship between robot count and system efficiency using throughput. Hence, we will only be comparing the number of robots of each system to determine the optimal number of robots to reach its maximum throughput.

5.2.2 Testing Result

Simulations were run for 2 hours to stabilize all the random events that might provide us inaccurate throughput. Figure 22 illustrates the graph between robot number and total throughput and Figure 23 shows the table of the number of robots with their respective throughput. From the table, the base case which is 500 robots is the number to compare the throughput with other numbers of robots. From the first few hundred increases of robots, it shows significant increases and slowly diminishing when the amount goes up. However, after reaching 2200 robots there is a slight decrease compared to 2300 robots, it may be the robots hits a saturation point. Even so, it is conceivable that the peak observed at 2,200 robots may partially arise from favorable randomization during that simulation run. This variability underscores the necessity for multiple simulation runs to average out random fluctuations. From 2300 robots onwards, the increase of the throughput is marginal. By plotting out the graph, we can see a logistic growth as the robot increases. This signifies that the system is reaching carrying capacity as there is limited space in the layout.

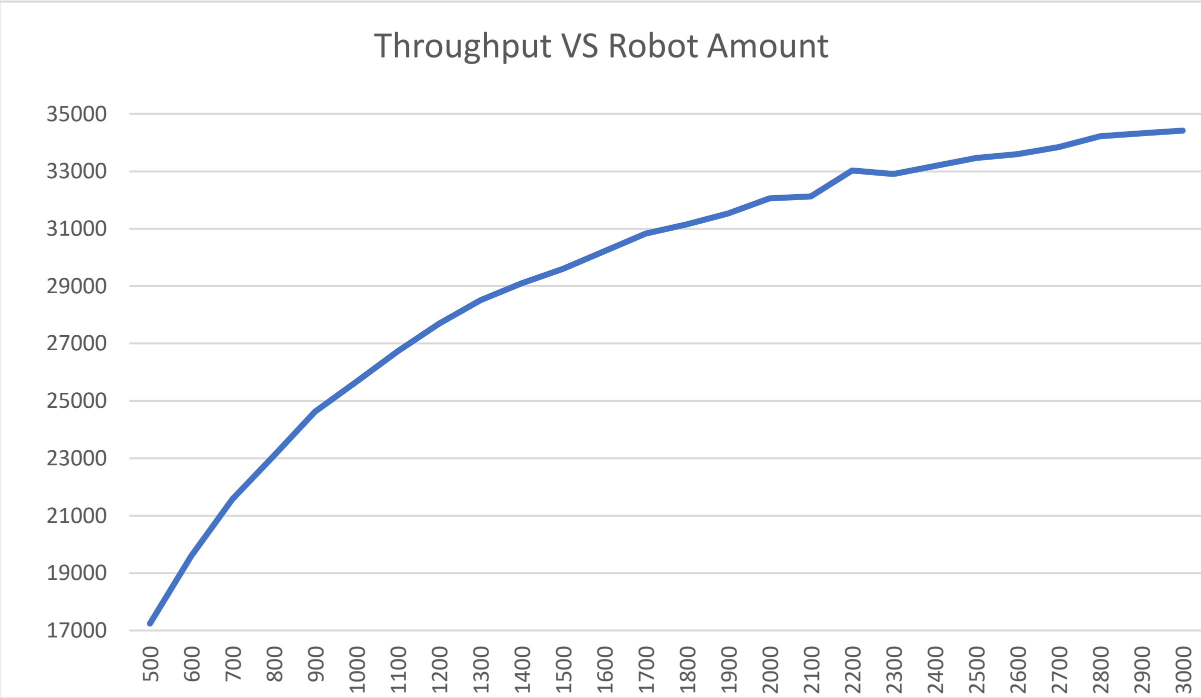


Figure 22 Throughput VS Robot Amount Graph

Robots	Throughput	Differences	Throughput Differential In Percentage Compared with 500 Robots
500	17241	N/A	N/A
600	19597	2356	113.6651%
700	21575	1978	125.1378%
800	23091	1516	133.9307%
900	24621	1530	142.8049%
1000	25675	1054	148.9183%
1100	26730	1055	155.0374%
1200	27691	961	160.6113%
1300	28510	819	165.3616%
1400	29098	588	168.7721%
1500	29598	500	171.6722%
1600	30215	617	175.2509%
1700	30826	611	178.7947%
1800	31152	326	180.6856%
1900	31536	384	182.9128%
2000	32054	518	185.9173%
2100	32130	76	186.3581%
2200	33024	894	191.5434%
2300	32901	-123	190.8300%
2400	33184	283	192.4714%
2500	33467	283	194.1129%
2600	33601	134	194.8901%
2700	33845	244	196.3053%
2800	34228	383	198.5268%
2900	34329	101	199.1126%
3000	34418	89	199.6288%

Figure 23 Throughput Differential Table

5.2.3 Testing Result Comparing with Previous Work

Since our project and [16] having different assumptions and different simulation time, therefore we are not able to compare our throughput with the number of robots. However, comparative analyses by determine the optimal number of robots required to reach maximum throughput can be compared using different assumptions and layouts based on both projects. Based on the results given by [16], we will be using their simulation that consists of ring road layout and mutual coordination. The peak throughput of [16] is 32199 with 900 robots in the layout. In order to having a fair comparison between the 2 projects, a predictive model was constructed based on [16] available data. The project provides 300 to 1000 robots range of throughput data whereas our project having a broader range of 500 to 3000 robots. Thus, polynomial regression is utilized to predict the throughput across different range of robot counts. The model is chosen as it simply able to capture non-linear relationships that typically happens in throughput data in a sorting centre and provides flexibility needed to fit various datasets. Figure 24 shows the predicted graph with the actual result of both projects.

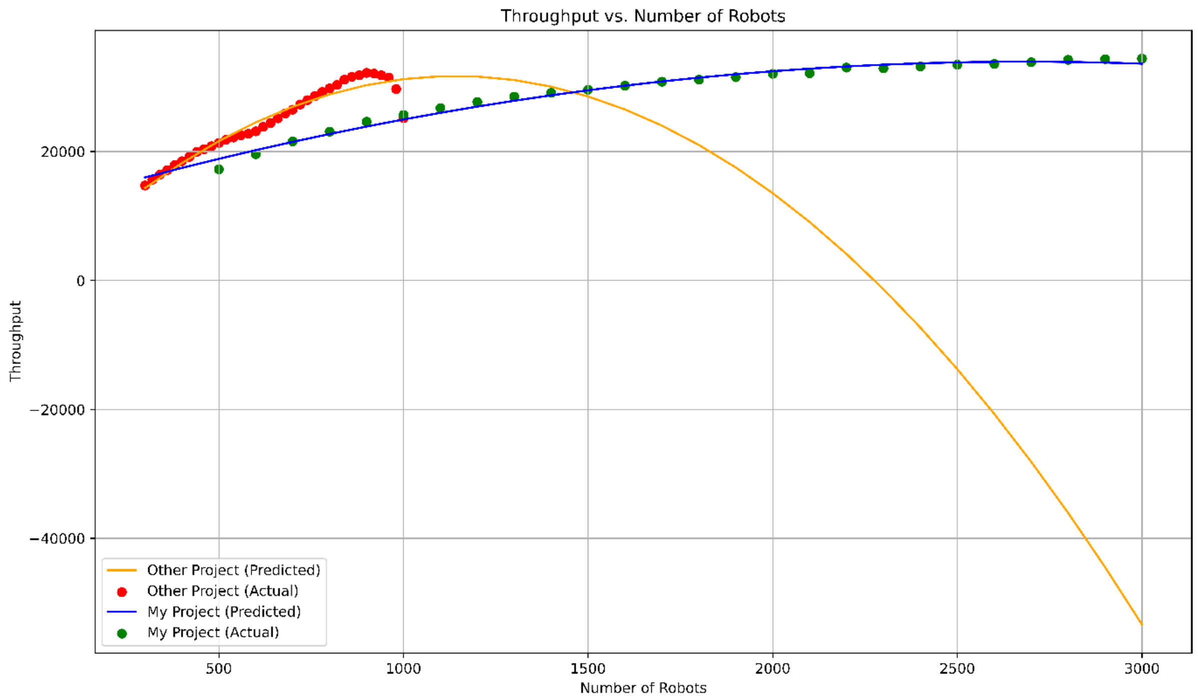


Figure 24 Previous Project vs Current Project

Based on the predicted graph, since [16] does not provide sufficient information after the decreasing of throughput, which is beyond 1000 robots, the number of throughputs drops below 0 after around 2200 robots based on the polynomial regression prediction model. However, the graph still presents the relationship between robot density and operational throughput in the layout. In the comparative project, an increase in robot numbers initially leads to enhanced throughput, up until a point of saturation. Beyond this threshold, the predictive model foresees a decline, indicating there might be potential deadlocks or unsolvable traffic in the previous project. In contrast, our project demonstrates a consistent level of throughput, implying that the layout can support a greater density of robots without a notable decrease in throughput, reflecting a more scalable and congestion-resistant system design. However, to consider that these predictions are model based, the comparative project should be further validated through empirical testing to confirm the layout's capacity to handle increased robot numbers without compromising on throughput.

5.3 Objectives Evaluation

The objectives of the project have been found out and achieved that indicates some advancements in the management of autonomous mobile robot (AMR) systems in the sorting centre.

The first objectives are to optimize cluster formations and dispersion. The focus on intersections as strategic points for forming and breaking clusters has proven effective. By utilizing token passing, it can easily decide to break/continue the clusters. Besides that, managing traffic flows at the intersections able to avoid traffic accumulations and smoother navigations for these robots to moves.

Secondly, path planning with token-based approach has been achieved. The implementation of a token-based algorithm managed by a central server shows effective on managing the movements of the robots. The approach adeptly avoids collisions and deadlocks which provide robustness of the path planning algorithms. Besides that, allocator that acts as a centralized control server ensures a harmonious flow of robot traffic.

Lastly, algorithm for token passing has been successfully implemented. The proposition and integration of a token passing algorithm among robots have been instrumental in elevating throughput. By allowing robots to pass tokens to those directly behind them, especially in high-density areas like intersections, the algorithm enhances traffic flow and operational efficiency. This feature not only mitigates potential deadlocks in intersections but also ensures that throughput is maximized with the cooperation with token-based management.

5.4 Concluding Remark

The testing results provide a compelling confirmation of the project's success in enhancing robot sorting centre system’s performance while also achieving all the objectives. The data indicates that with strategic management at intersections, robust token-based path planning, and token passing approach, the system meets the expectations in terms of throughput and efficiency. These results not only validate the effectiveness of the developed algorithms but also having the potential for scaling operations while maintaining the performance.

Chapter 6 Conclusion and Recommendations

6.1 Conclusion

To shortly conclude, this project developed an automated robotic sorting center built with frameworks and path-finding algorithms that navigates the robots to perform its task. The simulations are written with Python language using PyCharm. One of the core properties of this project it includes token-based approaches that avoids collisions between robots. Besides that, token passing approaches has been applied to freely construct and breaks clusters which the robots are moving together. The project discovered that prioritizing intersection grids able to effectively solve traffics. By doing various testing, we have found that it peaks once at 2200 robots with the total throughput of 33204. The throughput of the project increases marginally even after losing some throughput passing 2300 robots. This project tackles the challenges in the field of automated logistic and material handling among current sorting centers. The algorithms and layouts that are proposed could be an inspiration how sorting centers can operate that leads to a more agile and effective operations.

6.2 Novelty

The novelty of this work lies in the integration of a token-based approach and the strategic implementation of token passing within a sorting centre system. This innovative framework facilitates a coordinated, collision-free environment, where robots dynamically acquire and relinquish access to critical points, ensuring fluidity of movement throughout the system. Not only that, token passing mechanism further refines this process by allowing sequential handovers of movement to certain robots, significantly solving potential deadlocks and traffic that might build up. The combination of these strategies synergizes together, providing a layout that is highly scalable.

6.3 Recommendations and Future Works

From the implementation challenges, one of the works that can be accessed in the future is to further enhance the greedy allocator which assigns multiple robots towards the same location. Currently, the allocator was not matured enough to detect potential deadlock as robots might move to the same location back-to-back, resulting in blocking the exit of the robots. Thus, future iterations should aim to develop a more sophisticated allocation strategy. The objective would be to move away from the greedy allocator and toward an algorithm that considers the global state of the system. An allocator that can predict potential deadlocks on the pickup and drop-off point based on current and future allocations should be the future work that we are looking for.

Besides that, there is a clear need for a comprehensive review of the layout design used in the simulation. The future layout should balance pickup points and drop-off points and strategically place charging stations to minimize travel time. This might involve computational modeling and simulation to explore various layout configurations, aiming to identify the most efficient design based on different number of robots. Factors such as the average distance between pickup location and drop-off location and the likelihood of congestion area should be considered in the layout optimization process.

References

[1] Amazon, "What Is Ecommerce? Definition, Types, Advantages, and Disadvantages," Amazon, [Online]. Available: https://sell.amazon.com/learn/what-is-ecommerce#:~:text=An%20ecommerce%20business%20uses%20digital. [Accessed August 26, 2023].

[2] F. Kerick, "The Growth of eCommerce," [Online]. Available: https://medium.com/swlh/the-growth-of-ecommerce-2220cf2851f3. [Accessed 26 August 2023].

[3] I. Converge, "6 Key Warehouse Management Challenges for Manufacturers," [Online]. Available: https://www.itconvergence.com/blog/6-key-warehouse-management-challenges-for-manufacturers/. [Accessed 26 August 2023].

[4] L. Goodwin, "The 5 Challenges to AGV Implementation and How to Overcome Them," [Online]. Available: https://www.flexqube.com/news/the-5-challenges-to-agv-implementation-and-how-to-overcome-them/. [Accessed 26 August 2023].

[5] C. Factory, "Malaysia eCommerce Statistics and Trends in 2023," [Online]. Available: https://blog.commissionfactory.com/ecommerce-marketing/malaysia-ecommerce-statistics. [Accessed 26 August 2023].

[6] S. KEMP, "DIGITAL 2022: MALAYSIA," Datareportal, [Online]. Available: https://datareportal.com/reports/digital-2022-malaysia. [Accessed 27 August 2023].

[7] s. ebbers, "why is there a global labor shortage?," Randstad, [Online]. Available: https://www.randstad.com/workforce-insights/talent-acquisition/why-there-a-global-labor-shortage/. [Accessed 27 August 2023].

[8] S. Zahidi, "Future of Jobs Report," World Economic Forum, 2023.

[9] Q. G. J. C. P. Y. ZHENG ZHANG, "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classificatio," *IEEE Access,* vol. 6, pp. 26022-26035, 2018.

[10] G. Qing, Z. Zheng and X. Yue, "Path-planning of automated guided vehicle based on improved Dijkstra algorithm," pp. 7138-7143, 2017.

[11] Y. Zhao, X. Liu, G. Wang, S. Wu and S. Han, "Dynamic Resource Reservation Based Collision and Deadlock Prevention for Multi-AGVs," *IEEE Access,* vol. 8, 2020.

[12] T. a. X. Y. a. Z. D. Zheng, "AGV Path Planning based on Improved A-star Algorithm," pp. 1534-1538, 2019.

[13] T. K. S. K. J. L. S. K. Hang Ma, "Simulation-based Multiple Automated Guided Vehicles Considering Charging and Collision-free Requirements in Automatic Warehouse," pp. 1376-1380, 2018.

[14] F. C. Belkacem Khaldi, "An Overview on Swarm Robotics: Swarm Intelligence Applied to Multi-robotics.," *International Journal of Computer Applications,* vol. 126, 2015.

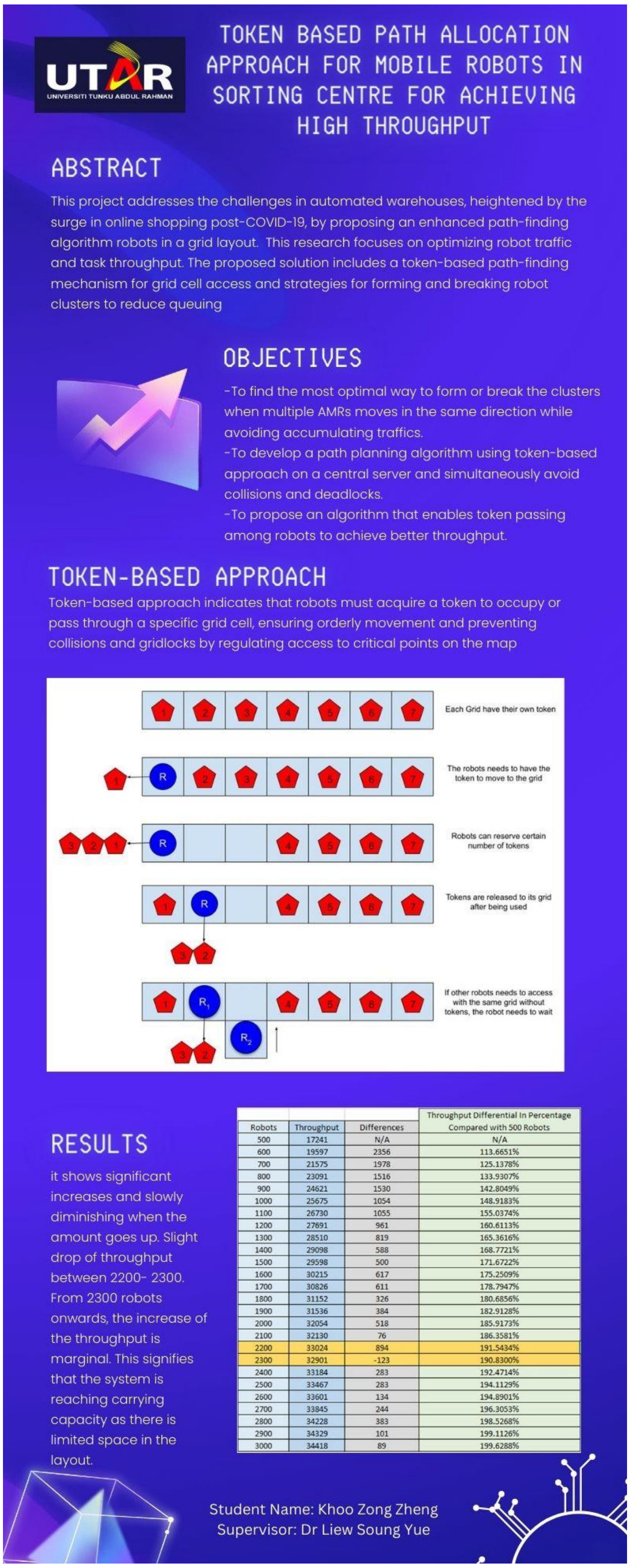
[15] J. L. T. K. S. K. S. K. Hang Ma, "Lifelong Multi-Agent Path Finding for Online Pickup and Delivery Tasks," 2017.

[16] J. Q. Biao, "Multi-Robot Path-Planning Algorithm for Automated Parcel Sorting Centre with Ring Road Layout," 2023.

Appendix

1. **1.** Code for Simulation in Python: https://github.com/BerryKhoo/FYP/blob/main/FYP2.py
2. **2.** Code for the polynomial regression predictive model: https://github.com/BerryKhoo/FYP/blob/main/FYP2%20Throughput%20Predictive%20Model.ipynb

Poster


TOKEN BASED PATH ALLOCATION APPROACH FOR MOBILE ROBOTS IN SORTING CENTRE FOR ACHIEVING HIGH THROUGHPUT
ABSTRACT
This project addresses the challenges in automated warehouses, heightened by the surge in online shopping post-COVID-19, by proposing an enhanced path-finding algorithm robots in a grid layout. This research focuses on optimizing robot traffic and task throughput. The proposed solution includes a token-based path-finding mechanism for grid cell access and strategies for forming and breaking robot clusters to reduce queuing
OBJECTIVES
-To find the most optimal way to form or break the clusters when multiple AMRs moves in the same direction while avoiding accumulating traffics.
-To develop a path planning algorithm using token-based approach on a central server and simultaneously avoid collisions and deadlocks.
-To propose an algorithm that enables token passing among robots to achieve better throughput.
TOKEN-BASED APPROACH
Token-based approach indicates that robots must acquire a token to occupy or pass through a specific grid cell, ensuring orderly movement and preventing collisions and gridlocks by regulating access to critical points on the map
Each Grid have their own token
The robots needs to have the token to move to the grid
Robots can reserve certain number of tokens
Tokens are released to its grid after being used
If other robots needs to access with the same grid without tokens, the robot needs to wait
RESULTS
it shows significant increases and slowly diminishing when the amount goes up. Slight drop of throughput between 2200- 2300. From 2300 robots onwards, the increase of the throughput is marginal. This signifies that the system is reaching carrying capacity as there is limited space in the layout.
Robots | Throughput | Differences | Throughput Differential In Percentage Compared with 500 Robots
500 | 17241 | N/A | N/A
600 | 19597 | 2356 | 113.6651%
700 | 21575 | 1978 | 125.1378%
800 | 23091 | 1516 | 133.9307%
900 | 24621 | 1530 | 142.8049%
1000 | 25675 | 1054 | 148.9183%
1100 | 26730 | 1055 | 155.0374%
1200 | 27691 | 961 | 160.6113%
1300 | 28510 | 819 | 165.3616%
1400 | 29098 | 588 | 168.7721%
1500 | 29598 | 500 | 171.6722%
1600 | 30215 | 617 | 175.2509%
1700 | 30826 | 611 | 178.7947%
1800 | 31152 | 326 | 180.6856%
1900 | 31536 | 384 | 182.9128%
2000 | 32054 | 518 | 185.9173%
2100 | 32130 | 76 | 186.3581%
2200 | 33024 | 894 | 191.5434%
2300 | 32901 | -123 | 190.8300%
2400 | 33184 | 283 | 192.4714%
2500 | 33467 | 283 | 194.1129%
2600 | 33601 | 134 | 194.8901%
2700 | 33845 | 244 | 196.3053%
2800 | 34228 | 383 | 198.5268%
2900 | 34329 | 101 | 199.1126%
3000 | 34418 | 89 | 199.6288%
Student Name: Khoo Zong Zheng
Supervisor: Dr Liew Soung Yue

Plagiarism Check Result

Turnitin Originality Report

Processed on: 24-Apr-2024 22:55 +08
ID: 2360442429
Word Count: 11018
Submitted: 1

Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput By Khoo Zong Zheng

Similarity Index	Similarity by Source	
10%	Internet Sources:	5%
	Publications:	7%
	Student Papers:	3%

2% match (Internet from 05-Jan-2023)
https://deepai.org/publication/lifelong-multi-agent-path-finding-for-online-pickup-and-delivery-tasks

1% match (Yunlong Zhao, Xiaoping Liu, Gang Wang, Shaobo Wu, Song Han. "Dynamic Resource Reservation Based Collision and Deadlock Prevention for Multi-AGVs", IEEE Access, 2020)
Yunlong Zhao, Xiaoping Liu, Gang Wang, Shaobo Wu, Song Han. "Dynamic Resource Reservation Based Collision and Deadlock Prevention for Multi-AGVs", IEEE Access, 2020

1% match (Guo Qing, Zhang Zheng, Xu Yue. "Path-planning of automated guided vehicle based on improved Dijkstra algorithm", 2017 29th Chinese Control And Decision Conference (CCDC), 2017)
Guo Qing, Zhang Zheng, Xu Yue. "Path-planning of automated guided vehicle based on improved Dijkstra algorithm", 2017 29th Chinese Control And Decision Conference (CCDC), 2017

1% match (Zheng Zhang, Qing Guo, Juan Chen, Peijiang Yuan. "Collision-Free Route Planning for Multiple AGVs in Automated Warehouse Based on Collision Classification", IEEE Access, 2018)
Zheng Zhang, Qing Guo, Juan Chen, Peijiang Yuan. "Collision-Free Route Planning for Multiple AGVs in Automated Warehouse Based on Collision Classification", IEEE Access, 2018

1% match (Tao Zheng, Yangiang Xu, Da Zheng. "AGV Path Planning based on Improved A-star Algorithm", 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2019)
Tao Zheng, Yangiang Xu, Da Zheng. "AGV Path Planning based on Improved A-star Algorithm", 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2019

< 1% match (Internet from 16-Dec-2023)
https://www.mdpi.com/2624-8921/3/3/27

< 1% match (Internet from 02-Jul-2023)
https://www.mdpi.com/1996-1073/15/23/8910

< 1% match (Internet from 16-Dec-2023)
https://www.mdpi.com/2077-1312/11/8/1556

< 1% match (Zheng Zhang, Qing Guo, Juan Chen, Peijiang Yuan. "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification", IEEE Access, 2018)
Zheng Zhang, Qing Guo, Juan Chen, Peijiang Yuan. "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification", IEEE Access, 2018

< 1% match (C. K. M. Lee, K. L. Keung, K. K. H. Ng, Daniel C. P. Lai. "Simulation-based Multiple Automated Guided Vehicles Considering Charging and Collision-free Requirements in Automatic Warehouse", 2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2018)
C. K. M. Lee, K. L. Keung, K. K. H. Ng, Daniel C. P. Lai. "Simulation-based Multiple Automated Guided Vehicles

WEEKLY LOG

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 2**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE Finished solving intersection issues that might causes deadlock.
2. WORK TO BE DONE Token Passing, Layout reconstruct (using previous project)
3. PROBLEMS ENCOUNTERED None
4. SELF EVALUATION OF THE PROGRESS On Pace

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 4**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE Finished Reconstructing the map layout.
2. WORK TO BE DONE Token Passing Approach
3. PROBLEMS ENCOUNTERED None
4. SELF EVALUATION OF THE PROGRESS On Pace

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 6**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE On phase solving token passing issues, yet to be done.
2. WORK TO BE DONE Completing token passing approach.
3. PROBLEMS ENCOUNTERED Not able to effectively pass the token at the intersection causing deadlock.
4. SELF EVALUATION OF THE PROGRESS On Pace

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 8**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE Finished all the layout/algorithm, completed token passing.
2. WORK TO BE DONE Testing simulation
3. PROBLEMS ENCOUNTERED None
4. SELF EVALUATION OF THE PROGRESS On Pace

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 10**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE

Testing and discovering the results of simulations.

2. WORK TO BE DONE

Finishing the simulations, result gathering and analysis

3. PROBLEMS ENCOUNTERED

None

4. SELF EVALUATION OF THE PROGRESS

On Pace

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT
(Project II)

Trimester, Year: Y3S3	**Study week no.: 12**
Student Name & ID: Khoo Zong Zheng 20ACB01908	
Supervisor: Dr. Liew Soung Yue	
Project Title: Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput	

1. WORK DONE Finished all the simulation, result has been analyzed and written into the report.
2. WORK TO BE DONE -
3. PROBLEMS ENCOUNTERED None
4. SELF EVALUATION OF THE PROGRESS Completed

Supervisor's signature

Student's signature

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 24/04/2024	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Khoo Zong Zheng
ID Number(s)	20ACB01908
Programme / Course	Bachelor of Computer Science
Title of Final Year Project	Token Based Path Allocation Approach for Mobile Robots in Sorting Centre for Achieving High Throughput

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: ___10____ % **Similarity by source** Internet Sources: 5% Publications: 7% Student Papers: 3%	Within the required range.
Number of individual sources listed of more than 3% similarity: 0	Within the required range.
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor	Signature of Co-Supervisor
Name: Liew Soung Yue	Name: ______________
Date: 26/4/2024	Date: ______________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB01908
Student Name	Khoo Zong Zheng
Supervisor Name	Prof Ts Dr Liew Soung Yue

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

____

(Signature of Student)

Date:24/4/2024