DEVELOPING A FLEXIBLE AND ADAPTIVE SHIPYARD FLEET FOR LAST MILE DELIVERY

Submitted to the College of Science and Engineering in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Robotics) (Honours), Master of Engineering (Electronics) at Flinders University – Adelaide Australia

Author: Richard Ellis

Supervisors: Giselle Rampersad & Russell Brinkworth

1 EXECUTIVE SUMMARY

As part of the Ship building program, Defence Company Australia is interested in implementing Autonomous Ground Vehicles (AGVs) and potentially aerial drones to move components and materials to various locations in the shipyard. This is intended to not only increase cost efficiencies but also improve safety for workers by way of reduction of heavy lifting.

Consequently, the objective of this project is to explore the development of a flexible and adaptive AGV system that can be applied to the shipyard once building locations have been confirmed.

It would address the key research question of 'how can autonomous vehicles be used for effective fleet delivery in an adaptive system allowing for last-mile delivery?'

To answer this question the study takes an option based approach to developing the shipyard AGV model, where various options are evaluated to see which option is more ideal.

by evaluating the different options and using the pros from previous options an adaptive shipyard model was successfully implemented where the drop off, pickup points and number of AGVs could be controlled. Overall, the system performed as expected and was able to replicate similar trends seen in other AGV models as well as the one from industry. However, the model is not as user friendly and still requires code elements to be adjusted prior to a simulation run.

Further development of the model would involve integrating pedestrians and a better user interface to provide quicker more reliable results between simulations as well as further investigation into the impacts of an AGV system in terms of human factors through interviews of current employees for which the system will work alongside.

Nevertheless, the study made a valuable contribution in addressing the key problem of how AGV fleet management systems can be applied in shipbuilding.

Disclaimer

I hereby acknowledge that in accordance with the University's policy on plagiarism, and unless otherwise referenced, all material presented in this report is my own.

Richard Ellis

June 2021

Acknowledgements

I would like to acknowledge the following for their assistance in producing this report:

- Giselle Rampersad and Russell Brinkworth for supervising and guiding me on this project.
- Defence Company for giving me the opportunity to undertake the project

2 TABLE OF CONTENT

3 TABLE OF FIGURES

4 TABLE OF TABLES

5 INTRODUCTION

5.1 PROJECT BACKGROUND

The purpose of this thesis is to explore the development of an adaptive AGV system that can be applied to various environments as to give the ability to determine the effects of increasing the AGV fleet size.

Part of the interest in this thesis is that Defence Company has been designated the Ship building program and are currently developing new warehouses and various other operations. Having an adaptive AGV system will allow for a way to see how changing various drop off or pickup locations can affect the outcome of fleet size and or overall delivery time of products.

5.2 STATEMENT THAT SETS THE SCOPE

The scope of the project involves the development of a method in which building supplies can be transported to various points on the shipyard autonomously while having the ability to change the number of transporters.

5.3 CURRENT SITUATION

The vision for The Defence Company project is to enable 24/7 delivery of materials for construction of the Ships reducing staff-hours in material handling and reducing the potential of injury due to lifting and manoeuvring of materials. The project will consist of 4 phases. Phase 1 is the planning phase during which, a proposal is created, which describes the method of approach towards the project, phase 2 is looking into the pervious literature on human factors and implementing AGV systems. Phase 3 is creating an adaptable AGV system and phase 4 is reviewing the results of the system.

5.4 IMPORTANCE OF THE PROPOSED RESEARCH

A significant factor in the application of AGV’s is to save money in the long term. This especially applies to Defence Company due to the lifetime of the Ship building contract. An

AGV system is capable of not only replacing the cost of forklifts and their operators but also running 24 hours a day, seven days a week, without human supervision. Cost per forklift is approximately $240,000. This cost includes the wages of three drivers, capital expenditure or leasing for the vehicle, and running costs. Over ten years, the cost of a forklift equates to approximately $2.4 million per forklift. The AGV will cost approximately the same for the first year. However, the costs will be reduced to $4,000 to $6,000 per year ('AGVs drive warehouses of the future' n.d.).

AGV's will also allow for greater safety to employees preventing not only muscle strain from moving objects constantly but by also providing a safer method of travel for material, preventing any accidental collisions with people or objects(Bostelman 2009), this in itself ads immeasurable value to the company.

5.5 THE PROJECT AIMS

The overall aim of the project is to create an adaptive model for controlling AGVs of varying fleet size while also being able to change both pickup and drop off locations in order to understand the impacts of varying AGV numbers on the amount of product that can be moved.

5.6 THE DEFENCE COMPANY REQUIREMENTS

Before the project can be developed, The Defence Company had various requirements that were found and spoken about in multiple meetings. These requirements were then placed into a MoSCoW diagram and expanded upon to apply clarity to each aspect of the project.

Must:

- Ability to select number of AGV
    - The ability to select the number of AGVs is crucial in the defence company's goal to estimate the fleet size necessary to deliver tier products through the shipyard. Having this knowledge directly impacts the cost of the system and reliability when factoring in maintenance schedules at a later date

- Select amount of resources
    - The number of resources delivered is key as it allows them to set up custom simulations of various days or predetermined scenarios that they would have encountered in previous projects giving them more agility.
- Select distribution centres
    - As the shipyard is still under construction, being able to select the distribution centres is a key aspect of the software as it allows the engineers at the defence company the flexibility to move things around to try and optimise their own systems. It also allows the system to adapt over time as new knowledge is gained about the shipyard.
- Track idle, delivering, and returning to base
    - These are the key pieces of information that will allow the defence company the ability to see how well the system is being utilised and if they have added too many AGVs or not enough. This however, will only take into consideration the system as a whole
- Recharge system when needed.
    - This is one of the most important aspects of an AGV as it determines when an AGV needs to go back to charge and when the AGV can be used again, this is crucial as AGVs have to manage their charging along with their allocations such that there is not task accepted that would put the battery in the red. Furthermore, sufficient charging stations will need to be available for the AGV to charge at.

Should:

- Every individual AGV data tracked
    - Having a system that shows the usage of each individual AGV will allow for greater accuracy when making the decision as to whether or not to increase or decrease the number of AGVs. This is because when looking at the overall utilisation of a system with, say, 20 AGVs, it will be hard to notice if a single

AGV is never being used. Thus, it would mean that the system has an added cost of approximately $250 thousand dollars that could be spent elsewhere.

- 3D implementation of the shipyard
    - Placing the AGVs in a known space not only provides realistic data to the simulation but also helps improve the human factor elements of the systems as those that interact with the system can see and visual the application with less bias and a clearer picture.
- Human Factors
    - The human factor is a crucial aspect of any project that is integrating automation and humans in the same process. This is because, at the end of the day, the system will be utilised by people and has to be accepted as a viable option by people without them fearing that automation will take over their employment.
    - Gaining input from various engineers and end-users will not only help make a better system but also prepare them for combined autonomy and what is potentially to come in the future.
- Cost of implementing AGVs
    - Cost is a major factor in any project; thus, having the costs of the AGVs recognised allows for the system to provide more cost-effective value to the defence company helping them achieve one of their core goals of trying to save money as they will be able to compare the system’s costs to their current procedures.

Could:

- Varying AGV models
    - To further add realism to the system allows the user to select various AGV models with varying factors that will affect the stem, such as top speed and load capacity. This will overall provide a more realistic system that is likely to provide an almost exact answer as to the number of AGVs required.
- Weather as a factor

    - Implementing weather as a factor provides more realistic simulation over the duration of days, weeks or months depending on the given task. Weather is a key component as the AGV will not be able to operate in extreme rain, thus reducing the operational days and increase the idle time of the system.
- Night-time vs daytime simulation model
    - Implementing night-time or daytime will allow for unique scenarios to be simulated, such as moving most components at night with lighter pedestrian foot traffic and then making smaller deliverers during the day.
- Pedestrians in the simulation
    - Pedestrians will increase the accuracy of the systems. It provides a dynamic obstacle for the AGV to navigate around, adding extra time to the system for both delivering and returning.

Will Not:

- Drones
    - During the earlier phase of the project, there was discussion around the implantation of drones in the system. However, due to the various safety hazards that this could cause, giving a shipyard is a dynamic and moving environment.
- The maintenance schedule factored into the cost.
    - The maintenance schedule will not be factored into the simulation, but instead, generic costs will be shown.
- Digital Twin
    - Due to the scale and scope of the project, there will be no digital twin implemented as this is more likely to be extra by the manufacture of the AGV system that the defence company would purchase and use.

5.6.1 ASSUMPTIONS

Assumption 1: Each AGV can move all around the layout

Assumption 2: The AGV only transports one kind of product as it just matters that the AGV is carrying something such as a pallet

Assumption 3: Weather has no impact on the AGV

Assumption 4: AGVs do not break down

6 LITERATURE REVIEW

The purpose of the literature review is to understand the current research that exists involving the current project. The review infers gaps in the current research and addresses how the project will fill these gaps. This literature review aims to answer the following questions:

- Is the investment of time and resources for the defence company to use AGVs to move components in the shipyard justified?
- What alternatives are there to AGVs in the industry?
- What current AGV technologies currently exist?
- Have there been any similar situations where AGVs have successfully been integrated into a shipyard environment
- What are the human factors of integrating AGVs into the shipyard?

According to (Ramirez-Peña et al. 2020), one of the largest factors that can impact shipbuilding and contribute to bringing shipbuilding into an industry 4.0 framework is the logistics supply chain. Lean, Agile, Resilience and Green to define what the Shipbuilding Supply Chain should be.

6.1 INDUSTRY 4.0 SUPPLY CHAIN

One of the biggest changes in the industry is the shift from computers to smart devices and utilising infrastructure based on cloud computing (Tjahjono et al., 2017). Industry 4.0 is playing a large role in how production shop floors currently operate and include the use of innovative developments in the digital age of technology, including advanced robotics and artificial intelligence, data capture and analytics, digital fabrication such as 3D printing, platforms that use algorithms to direct autonomous vehicles to mention a few. (Schrauf n.d.). According to (Kiel et al. 2017), one of the biggest challenges with industry 4.0 is the technical integration of systems. This is due to the integration and infrastructure upgrades between many different systems. A difficult factor when integrating industry 4.0 is predicting the type and amount of infrastructure upgrades needed.

One of the main challenges that the defence company will be facing is a lack of technological infrastructure in the current shipyard. This lack of infrastructure is the most pressing challenge when integrating new machines and methods (Moktadir et al. 2018). Thus, to mitigate this lack of infrastructure, a key element would be smart mobility (‘ch-en-manufacturing-industry-4-0-24102014.pdf’ n.d.).

6.1.1 THE INTERNET OF THINGS

The internet of things (IoT) is a concept which describes how various devices are connected (Sisinni et al. 2018). IoT is widely used in healthcare, utilities and transportation (Sezer, Dogdu & Ozbayoglu 2018). Using IoT, the end-user/customer moves closer to a plug and play environment making it easier to operate and configure logistics remotely (Shafique et al. 2020). There are four stages to an IoT Architecture, as shown below in Figure 1.

Removed due to copyright restriction

Figure 1 IoT Architecture showing the various stages of devices connecting to the internet

The stages are made up of Sensors and actuators, Internet getaways and Data Acquisition Systems, Edge IT and Data centre and cloud (Stokes 2019). Some of the major concerns that cannot be overlooked when integration an IoT system is cloud Attacks and security vulnerabilities (‘7 Big Problems with the Internet of Things’ n.d.). This is especially an area that the defence company will consider when trying to integrate an automated delivery system that can function in an indoor and outdoor environment. Due to this type of security risk being outside the project's scope, it will be assumed that the AGV network is already secure and safe from any external forces of attack.

6.1.2 BIG DATA

One of the major benefits of integrating industry 4.0 is the amount of data captured by various sensors and systems. Big data is a term for massive data sets displaying large amounts of varied and complex structures with the difficulties of storing, analysing, and visualising results (Sagiroglu & Sinanc 2013). The data captured allows organisations to gain deeper insights into their business (Davenport, Barth & Bean n.d.). In the case of the defence company, this data can be used to predict and optimise the factory's logistics. There are three main components to Big data, namely variety, velocity and volume (Sagiroglu & Sinanc 2013). Variety makes big data big, volume or the size of data, is larger than terabytes and petabytes, and velocity is how quickly the data can be captured and stored. The problem that can occur when implementing Big data is collecting unnecessary data, taking up computer processing time, and server space. Data should only be collected when a method of analysing said data is known.

According to (Mourtzis, Vlachou & Milas 2016), there are different levels of a manufacturing enterprise. The lower level generates data directly from machine tools and operators. This level of data is the most important and holds the most potential for manufacturers when analysed correctly.

6.1.3 SIMULATION

One of the most indispensable tools for digital and advanced manufacturing in this modern era is computer-aided simulations (Mourtzis, Doukas & Bernidaki 2014). Simulations are a data visualisation method that allows business to better understand the dynamics of adding upgrades to their facilities (Rodič 2017). Simulation modelling helps reduce costs, shorten development cycles as well as increase the quality of manufacturing and its processes.

Simulations are often used in situations where a mathematical model cannot be used to resolve an issue. This is because simulations can be altered more simply by combining multiple mathematical models in a visual fashion, as shown below in Figure 2.

Removed due to copyright restriction

Figure 2 Factory simulation model demonstrating what a 3D simulation environment can be

Using simulation, a business can gain insights into complex systems allowing the testing of new concepts or systems, resource policies and new operation before its real implementation (Mourtzis et al. 2015).

However, one of the major drawbacks in simulations is that an in-depth understanding of all the factors involved is needed to simulate something. Without the understanding, a simulation is likely to be flawed (‘Advantages and disadvantages of simulation - Computer simulation - GCSE ICT Revision’ n.d.). In order to mitigate this, a combination of real and simulated hardware known as Hybrid System Simulation is required. This process is when Real devices, like the machine tool or controller, are linked to a machine's simulated model to test the machine's behaviour during manufacturing the system. This method is used to try and create the most realistic scenarios that a machine will undergo (V G & Patil 2015).

In the case of the defence company, the MIR robot can be used to fine-tune the simulation by making sure the simulation produces similar times to that of real-world testing done by the MIR.

6.2 INDUSTRY WORKERS TRUST IN TECHNOLOGY

6.2.1 PHYSICAL

6.2.1.1 MANUALLY REPETITIVE TASKS

One of the main aspects that industry 4.0 and automation will have a large impact on is reducing manually repetitive tasks done by humans. The lead to reduce repetitive tasks comes from companies and organisations to improve the process in their business (Leopold, van der Aa & Reijers 2018). However, this automation can currently only occur in tasks that have limited flexibility and do not require advanced problem solving (Fantini et al. 2016).

In order to introduce robotics that is designed to take over work from humans for a manually repetitive task, large amounts of research and work have been placed into the design of collaborative robots, as shown below in Figure 3 (Djuric, Urbanic & Rickli 2016).

Removed due to copyright restriction

Figure 3 Collaborative Robot and Person Working Together Safely

Using hybrid teams of robots and humans, robots will compensate for the physical limitations of human workers, where robots help workers lift heavy items or take over other physical tasks (Pearce et al. 2018). The next evolution in collaborative robotics is the integration of intelligent systems that can adapt to the current employee working with the machine and complement or augment the capabilities of the human operator (Fletcher et al. 2020).

6.2.1.2 COLLABORATIVE HUMAN-MACHINE SAFETY

A key factor when implementing machines that can move 100's of Kgs to tons around is safety, specifically with the collaborative robots due to the close working proximity to humans (Chemweno, Pintelon & Decre 2020). Due to this, ISO 15066 has proposed guidelines as to which designers, integrators, and users need to consider when embedding passive and active safeguards on the robotic system as well as while designing collaborative areas within the work environment (14:00-17:00 n.d.). According to (Vysocky & Novak 2016), one of the main aspects of safety in collaborative robots is not so much the robot but the tools a robot uses. This means that depending on the given task of the collaborative robot, different safety procedures need to be firmly in place. (Grahn, Johansen & Eriksson 2017) It is recommended that collaborative robots be implemented safely. One approach that can be used is to simulate the environment in which humans and the robot will interact. Further steps then can be taken where hands-free gesture controls can be used to remove the human from the robot's path but still allow a collaborative environment where the robot compensates for the human and the human can perform the advanced problem solving that may be required (Nuzzi et al. 2019).

6.2.1.3 IMPROVING INTERNAL LOGISTICS AND TRANSPORTATION

Robots are limited to a factory's workstations, as discussed previously. Another area for collaborative robots is logistics and transportation. Often the human factor in logistics is undervalued and rarely considered (Grosse et al. 2017). A key element with dealing with transportation logistics is that there are different risk factors with inbound and outbound logistics, as this is a key issue as to when automation should or should not be applied (Cimini et al. 2019). The key areas of research seem to focus on inbound logistics, particularly in warehouses (Dewa, Pujawan & Vanany 2017). The key area where human factors and industry 4.0 collied in the warehouse is as mentioned before pertains to where movements are particularly repetitive, require intensive labour and are time-consuming (Koster, Johnson & Roy 2017). However, adding robots to reduce fatigue on works is not the only risk with human factors. Another risk is trust. According to (turvoinc 2017), the main reason for fear and trust in automation is the potential of job loss.

6.2.2 ORGANISATIONAL

6.2.2.1 BRIDGING THE GAP BETWEEN HUMANS AND MACHINES WITH HYBRID PRODUCTION ENVIRONMENTS

With the way that industry is currently going, it can be seen that the hybrid production systems rely on close human-machine collaborations (Shi, Yang & Jiang 2021). Using hybrid productions systems allow organisations to recoup organisational losses by leveraging the ingenuity of people, mobile robots, expert knowledge and automation (Becker & Stern 2016). Due to new interactions between workers and machines that will occur more often in future, new management methods can be used, thus replacing older, less efficient methodologies for managing people (Hecklau et al. 2016). This, however, is still in the early stages of research as there are many processes that still need to be addressed, such as how to capture the data and process information between the interaction of people and machines at a work station on a higher level allowing for true value added work to be captured such as decision-making and problem-solving skills , creativity, social behaviour and influences, and accounting for workers' unique abilities and characteristics (Gorecky et al. 2014).

6.2.2.2 HUMAN-MACHINE INTERACTIONS WILL AFFECT WORK ORGANISATION AND DESIGN

One of the key aspects when dealing with human-machine interactions is determining the various tasks that humans and robots will do and making a clear division of it (Kadir, Broberg & Conceição 2019). By doing this allows for greater optimisation within the interactive working environment. According to (Fantini et al. 2016) because of the nature and complexity of creating human and machine work environments a key factor for an organisation is to try and build up a positive environment for workers to maintain wellbeing. This is also considered an area that requires more research as there are many factors when dealing with humans when adding automated robotic components to their work environment that can be seen as a threat to their job (Złotowski n.d.). Thus, only time will tell how the future factory will operate as organisations continue to upgrade and research current combined manufacturing techniques.

6.3 ROBOTICS IN ASSISTIVE MANUFACTURING

6.3.1 KEY ISSUES WITH ROBOTICS IN ASSISTIVE MANUFACTURING

6.3.1.1 SAFETY AND MAINTENANCE

When a system has only a small amount of AGVs, AGV failure will not cause major congestion. The AGVs can also be easily changed with an AGV from backup. However, as AGV systems grow and move to a larger scale, many industries are starting to use the technology (Vis 2006). When increasing the number of active AGVs on a site, system malfunctions can start to cause serious congestion and restrict travel routes. This means that safe and reliable use of AGVs is crucial to a business to assure the efficiency and productivity of such kinds of AGV systems.

Due to the impact that AGV failure can have on a system, there has been a large focus of research into the improvement of AGVs. Such research includes the safety requirements and safety functions for a decentralised, controlled AGV system (Trenkle, Seibold & Stoll 2013). Another example of further research is conducted by (Krnjak et al. 2015), where the paper investigates an algorithm that prevents the occurrence of deadlock and lives lock situations. In a paper by (Ebben 2001), a method that was designed to manage failure for an underground transport system is investigated. A few failure scenarios in the research used system recovery, equipment failure, AGV failure, and repair as the main focus points. From the research, a method of using a separate AGV to move the failed AGV away by means of a tugging system was suggested. This method removes the potential hazard from the scene and tasks the AGV to a designated repair location. The paper further showed that implementing AGVs that will wait on standby help to maintain the ideal performance of the AGV system.

(Tavana, Fazlollahtabar & R. 2014) created an algorithm optimising an automated factory using time and cost measures. The algorithm uses a neural network to define suitable weights for time and cost objectives. (Tavana, Fazlollahtabar & R. 2014) The reliability of the manufacturing process as a method for figuring out the cost function by looking at the steadiness and stability of the system should be considered, however by using this method, it is not possible to determine the root cause of AGV failures as these faults can exist outside of manufacturing.

In the paper ('Improving safety and efficiency of AGVs at warehouse black spots - IEEE Conference Publication' n.d.), there is a focus on the safe operation of AGVs at warehouse blackspots such as intersections and how the AGVs decelerate when approaching these. The main issue the paper is addressing is how the black spot is often void of other vehicles or workers, thus making the deceleration often unnecessary. In order to maintain a safe and efficient working environment, the paper suggests adding an environment perception system consisting of one or several laser scanners in combination with a data processing electronic control unit (ECU). By adding this type of environment, the AGV system can be considered safer as the AGV can rely on its onboard sensors plus external sensors to manoeuvre corners.

(O'Connor 1994) wrote that the system's reliability could be calculated by the probability that the system is fully operational over a period of time. Thus it can be seen that an AGV maintenance schedule should be put into place. (Németh et al. 2019) created a simulation using Siemens Plant Simulation as part of the simulation software, an AGV maintenance schedule is included in the simulation run. Currently, the two most widely adopted maintenance options in industry are preventative and corrective maintenance, where preventative maintenance is conducted periodically. Corrective is conducted upon failure ('Maintenance Theory of Reliability, Springer Series in Reliability Engineering by Toshio Nakagawa | 9781852339395 | Booktopia' n.d.). of one of the major consequences when running simulations is that either method has implications for safety and for time output, this area, however, lacks study due to the fact that most manufactures supply their own maintenance thus taking away this responsibility from the business getting the AGVs installed (Lei et al. 2010).

6.3.2 ROBOTICS IN SHIPBUILDING

Shipbuilding is a key part of the Australian defence industry conducted here in South Australia. It is of strategic importance for Australia due to the nature of the current economy ('defence-briefing-note-oct17.pdf' n.d.) as the shipbuilding and other naval projects generate long-term employment for Australian workers. According to (Andritsos & Perez-Prat 2000), one of the most promising fields of improvements for shipbuilding is the integration of modern technology in the manufacturing process.

According to ('4 Top Robotics Solutions Impacting The Shipbuilding Industry' 2019), four main areas are impacting the current automation in the shipping industry. Welding & Cutting Robotics, Painting Robotics, Wearable Robotics & Exoskeletons and Collaborative Robotics. These automations are great; however, they require that the employees working in similar areas require a certain level of skill to understand and troubleshoot any faults or unpredictability in the machines (Klumpp 2018).

6.3.3 METHODS OF TRANSPORTING GOODS

According to (Romaine 2018), one method used to transfer raw materials, work-in-process, or finished goods is conveyor belts. Conveyor belts have been around since the 1800s, according to (Dijkhuizen 2017). During this time, there have been many advancements. Conveyor belts are designed to systematically carry and transport material, typically in an industrial or controlled environment ('How Do Conveyor Belts Work? | Belt Functions, Uses & Applications' 2019). Whilst this method is good at moving large quantities of products around a production site, there are drawbacks with the use of them (FERGAL 2018). Implementing Conveyor belts would require large amounts of space and lack portability, with the cost being almost twice as much as implementing mobile robots. This outcome is undesirable as large amounts of logistics are involved in manufacturing and moving large components that rely on a flexible infrastructure.

Another alternative to AGVs is forklifts. Forklifts have lower upfront cost as well as the ability to move over uneven surfaces easily while operating at faster speeds than AGVs ('Forklifts vs. Automated Guided Vehicles' n.d.).

Furthermore, drones are yet another vehicle that can be used in the logistics process of moving materials around the shipyard. Drone delivery is currently a sharply divided topic among industry professionals as some see it as a stunt and others as a game-changer ('Mobile Robots and Drones in Material Handling and Logistics 2018-2038: IDTechEx' n.d.). Drone deliveries largest problems are that it offers limited productivity in comparison to traditional means of delivery as drones require constant battery recharging with limited distances and trips (Shavarani et al. 2018). Due to this and the fact and that constant large structures are moving in a shipyard, causes potential safety hazards for use of this method (Korman 2019).

All three methods described above provide a means of moving products from one area of the shipbuilding facility to the other, however, as shown, each method provided implies either an increase in cost, increase in logistic operations or increase in safety hazards. According to (Schneider n.d.), it is still within a company's best interest to automate as it will lower costs as well as increase production quality. Thus, it is considered a necessary step to attempt to add automation to the shipbuilding process.

6.3.3.1 AGVS

6.3.3.1.1 THROUGHPUT

When working with automation, specifically AGVs, one of the key aspects is the job throughput, which is a value that is measured with respect to quantity and time. (Srinivasan, Bozer & Cho 1994) discuss a general-purpose model that can approximate the throughput capacity of a material handling system used in a manufacturing setting. Using the model developed by (Srinivasan, Bozer & Cho 1994), the throughput can be approximated prior to running simulations.

In research conducted by (Peters & Yang 1997), a modified algorithm is used to solve the layout design problem. In order to do this, the paper suggested a network flow formulation to determine the number and location of shortcuts for the inter bay transport system. This is shown below in a spine layout in Figure 4. By using this design layout, costs are reduced considerably.

Removed due to copyright restriction

Figure 4 Spine configuration for where AGVs can travel in a simple factory

Another method proposed by (Mousavi et al. 2017) is a mathematical model that was developed and integrated with evolutionary algorithms to optimise the task scheduling of AGVs with the objectives of minimising makespan and number of AGVs while considering the AGVs' battery charge with The hybrid GA-PSO algorithm producing the best result.

Another study by (Lyu et al. 2019) looks at the AGV scheduling problem in a flexible manufacturing system. This is done by simultaneously considering the optimal number of AGVs. The main difference between this study and previous ones is that it does not initially assume a fixed number of AGVs. It also factors in various routing problems and transport time. In order to do this, the algorithm is genetic-based combined with the Dijkstra algorithm that is based on a time window.

(Digani et al. 2019) offers a solution that uses ad-hoc pre-defined roadmaps. What this means is that the algorithm attempts to reduce the time that the AGV spends moving through complex traffic situations, thus maximising the overall throughput of the system. However, the research report does make the following assumptions: No external obstacles (e.g. people, manual forklifts), the velocity along a segment is constant, each segment can be occupied by at most one AGV, and each AGV has a different pair of initial and final positions–The algorithm by (Digani et al. 2019) provides optimal paths, as shown below in Figure 5:

Removed due to copyright restriction

Figure 5 AGV Path Maps From (Digani et al. 2019)

6.3.3.1.2 UNIT LOAD

A unit load is the number of items arranged or packaged as a single unit, typically on a pallet, to facilitate handling, transportation, and storage (‘What does unit load mean?’ n.d.). Thus, it is a key component in cost reduction as the greater the unit load of an AGV the few trips required to achieve a given task (Kuo, Krishnamurthy & Malmborg 2007). (Kuo, Krishnamurthy & Malmborg 2007) states that the unit load significantly impacts the operational efficiency of an AGV-based manufacturing system, especially as global supply chains have seen a significant increase in the use of automation. This is also supported by an older paper written by (Egbelu 1993).

The unit load is of significant importance as it directly affects the overall productivity of a manufacturing plant. Thus, in its nature has a large area of research dedicated to it. The main focus of the research is to obtain the best unit load size using tools of mathematical programming, heuristics and simulation. Modern operation research or integration of methods will further assist in being useful for major FMS types of industries to determine unit loadings.

One common example of the unit load problem is known as the knapsack problem. This is because the knapsack problem is an optimisation problem where a constrained number of items can fit in a backpack and each item has a value and a weight. The aim is to fit as much value into the knapsack without breaking the knapsack weight constraint ('What is the Knapsack Problem? - Definition from Techopedia' n.d.). This problem is relevant to the unit load as a common aspect of warehouse management is to-do with the collection of resources within the given AGV weight range. However due to the style that the defence company is trying to achieve these components would be preselected prior to distribution of the AGV, which will not be utilised as part of this thesis. As per discussion with the defence company the system will likely use whole crates which will have pre designated components inside the given crate.

In order to determine a unit load size, a rule of thumb that is used is to adopt a wide-shallow warehouse layout when the item demands are approximately equal, whereas a narrow-deep layout is preferred when the demand curves are steep (Guo, Yu & Koster 2016).

Looking at research from (Egbelu 1991), a method for selecting the unit-load size is that for each value of unit-load size generated by the search routine, the number of unit loads from a batch size is calculated. For each unit-load size generated, a corresponding minimum manufacturing time for the batch is determined. By implementing a method like this, the defence company will have a high chance of optimisation as the nature of shipbuilding requires different unique parts that by using a model that optimises the number of parts and delivery time of the parts would be ideal.

In a research paper by (Mahadevan & Narendran 1992), integer programming formulation of the problem of finding the optimal unit load size is analysed. The papers use an existing analytical model to decide the number of AGVs required. The model is created by

(MAHADEVAN & NARENDRAN 1990). The algorithm for the paper reduces the cost by using a function that has inputs of the sum of the costs of the MHVs and the pallets. With more modern methods as covered by (Riazi, Bengtsson & Lennartson 2020), the main focus for cost reduction in AGV systems is makespan, maximum lateness, and the sum of tardiness for an AGV system, as well as the energy consumption of the AGVs. The study was able to determine that cruise velocity and travelled distances is the main factor in energy costs. Thus, it was determined that optimising the productivity-related performance criteria also reduces energy consumption through less travelled distance.

6.3.3.1.3 FLOW PATH DESIGN

The flow path design for AGV systems is extremely important. The layout of the system will determine the overall distance that the AGVs will travel. The main objective when designing an AGV flow path is to try and minimise the total travel of loaded vehicles (KASPI & TANCHOCO 1990). The first flow model was formulated by (GASKINS & TANCHOCO 1987). The paper described the use of a zero-one integer programming problem. The paper does not simulate anything and describes that a simulation would be best to determine a flow path. The mathematical model does, however, allow for quick analysis of a general solution. (KASPI & TANCHOCO 1990) use a computationally efficient procedure that is based on the branch-and-bound technique. The model described has the purpose of finding the direction of flow for each section within network's flow path. The flow path network represents the pickup and delivery stations, aisles and aisle intersections, within a manufacturing plant. The paper by (Kaspi, Kesselman & Tanchoco 2002) provides an improved algorithm where only the intersection nodes of the network are included in the branch-and-bound algorithm. The main objective of the model is to set directions to the arcs in an undirected graph which represents the flow path network. This is done in order to show that the total vehicle travel distance is minimised. The from and to flow matrix includes both empty and loaded vehicles traveling at full capacity. This research aims to reduce the total travel distance. Initially research focused on the travel of loaded vehicles only but later extended to the empty vehicles too.

However, as AGV technology has progressed, so has flow path design. (Yan, Dunnett & Jackson 2018) found that; a Coloured Petri Net (CPN) method is a valid method for conducting performance assessment and route evaluation of multi-AGV systems.

Five CPN Systems in the paper are as displayed below:

1. *Path Petri nets (PPN) – for describing the layout configuration of the system.*
2. *Master Petri nets (MPN) – for governing the mission progress or phase change of individual AGVs in the system.*
3. *Recycle Petri nets (RPN) – for describing the recycling process of failed AGVs.*
4. *Corrective maintenance Petri nets (CMPN) – for defining the corrective maintenance of failed AGVs in the system.*
5. *Periodic maintenance Petri nets (PMPN) – for defining the periodic maintenance of all AGVs in the system.*

The systems, even though they are described individually, still pass data through the system.

In a paper written by (Sedehi & Farahani 2009), an algorithm for calculating the flow path design is used with a focus on single-loop AGV systems. The proposed algorithm determines the block layout, AGV single-loop flow path and pickup delivery stations simultaneously.

A Literature Review on Material Handling using AGV:

6.3.3.1.4 FLEET SIZE

Fleet size is the number of AGVs that will be used to transfer various good in a manufacturing facility ('6.4 Optimising Vehicle Size and Fleet Size' n.d.). According to (Tao Yifei et al. 2010), there are a few factors that affect how many AGVs are used. The main one is the layout of the guided path. Secondly are the location of the load transfer points and the type of strategy used for dispatching vehicles. The aforementioned paper mainly revolves around using queuing software for an analysis of the number of AGVs needed. (Egbelu 1987) proposes a few methods which are simple, rough cut, hand calculation method, this type of method is only useful at the early stages of evaluation as it can only factor theoretical results and cannot understand routing.

In a paper by (Rajotia, Shanker & Batra 1998), a model is created to obtain the optimal fleet size by determining the time an AGV spends in a system using three points, the origin point, the load pickup point, and the unload delivery point. Using this mathematical model and parameters were used to determine the number of AGVs needed. This allows for a good starting point without the use of simulation.

In a study conducted by (Fethi & Mehdi 2019), three networks were tested using ARENA software as previously described. It showed that there is a threshold limit for when AGV Fleets reach their maximum point where adding more into the system will either see no change in productivity or decrease productivity.

(Vivaldini et al. 2015) looked at a method where the complete material handling task is a journey mapped such that the minimum number of AGVs required to do the task is found. This can be approximated by adding the travel times (loading, unloading and waiting time) divided by the available time of the AGV.

6.3.3.1.5 MOBILE ROBOTS WITH MATLAB

MATLAB is one of the most used programming platforms designed specifically for engineers and scientists. MATLAB has tools that can be used to analyse data, develop algorithms and create models and applications. MATLAB has a new robotic system toolbox that now incorporates algorithms with a focus on mobile robotics or ground vehicle applications. The Toolbox allows for the entire mobile robotics workflow from mapping to planning and control to be created (‘Mobile Robot Algorithm Design - MATLAB & Simulink’ n.d.).

(Mahulea & Kloetzer 2015) created a free program that allows for a robot to do simple path planning motion control and modelling, using a Graphical User Interface (GUI). Using the software is a good way of getting an idea as to some of the capabilities of the MATLAB Robot toolbox.

(‘MARS - Multi-Agent Robot Simulator’ n.d.) is another software that has been created that simulates a powerful platform for mobile type robots in structured and unstructured environments. It simulates a team of robots moving in a 2D workspace. Both holonomic and non-holonomic vehicles are simulated. The model created by (Casini & Garulli 2016) is used for Simultaneous Localisation and Mapping (SLAM) or path planning in structured or unstructured 2D workspaces.

Further work that has been done in the MATLAB simulation environment is Control and Simulate Multiple Warehouse Robots (‘Control and Simulate Multiple Warehouse Robots - MATLAB & Simulink’ n.d.). The example shows a system that simulates multiple robots working in a warehouse. The simulation uses state flow figures where a central scheduler can send various commands to the robots.

6.3.3.1.6 A* NAVIGATION

A* navigation is one of the best methods for finding the shortest path (‘A* Search Algorithm’ 2016). A* is a graph and path travels search algorithm. The algorithm works by using a system of weights. Where the goal is to find the most cost effect path i.e. the lowest weighting which is likely to be the shortest path.

The A* algorithm works by keeping a tree of paths that originate from the start node, which is where the object is when the algorithm starts. The algorithm iterates through its main loop trying the most cost effect path each time, the loop will run until the algorithm reaches the main goal of the system (‘A* search algorithm’ 2021). Figure 6 below shows a representation of this algorithm (‘Introduction to A*’ n.d.).

Removed due to copyright restriction

Figure 6 Demonstration of A* navigation algorithm from the red dot to the blue dot

The line in Figure 6 above represents the shortest path and the high lightest squares are everywhere the algorithm checked.

6.3.3.1.7 COLLISION AVOIDANCE

In a paper written by (Chia-Han Lin & Ling-Ling Wang 1997) a fuzzy approach to collision avoidance was used. The aim of the paper is to find the optimal configuration of the minimum number of sensors that provide the robot with the clearest view. The initial model that they used 24 sensors mounted on a rectangular AGV this results in a sensor every 15°. This paper focuses more on the aspect of what the robot can see however in a paper by (Yan, Zhang & Qi 2017a) the Collison avoidance occurred when the AGV is route finding, this is because the system that is developed prevents dead lock thus when an AGV has found its route it blocks off other AGVs from using certain points in its route preventing the robot from common inContact with other AGVs thus there is no need to navigate around them.

In the paper "Collision avoidance among AGVs at junctions" by (Arora, Raina & Mittal 2000) a unique approach is taken to avoid collisions in a unidirectional system by incorporating 2 controllers, the controllers for the intersection will be chosen depending on the scenario when the AGVs approach the intersection.

The most applicable approach for this research topic will be the one by (Chia-Han Lin & Ling-Ling Wang 1997) as it focuses on the sensors of the system, since the system that is being developed does not really on a flow path the method of locking routes depending on scenarios does not apply. Thus, the AGV will need to make navigation decisions based on the current width it has to move in and the speed at which it is traveling along with the distance from the object. This means that having optimal sensors is ideal for free roaming AGV as it will allow the AGV to have the greatest amount of freedom and safety.

6.3.3.1.8 COST

Table 1 below shows the cost of an AGV broken down into five sections, preventative maintenance, corrective maintenance, AGV purchase cost based on the type of AGV, installation Costs, and running costs. These costs are not 100% accurate of the system but can give an approximate value of the system cost for the initial purchase and set up and yearly running costs.

Table 1 Costs of a two AGVs for 1 year

Maintenance Per AGV		
Type	Amount	Unit
Preventative Maintenance		
Number Per year	3	yr
Number Technicians	2	
Total Working Hours Per Maintenance	16	hr
Cost Per Technician	140	hr
Spare Parts	20000	$
Corrective Maintenance		
Number Per year	2	
Number Technicians	1	
Total Working Hours Per Maintenance	8	hr
Cost Per Technician	140	hr
Travel Cost	900	$
Total Cost	$16,580	Yr
Total Cost 5 years	$102,900	5/yr
AGV Purchase Cost Based on Type		
Automated guided carts (AGCS)	38000	$
Automated tow tractors	76000	$
Automated forklifts	150000	$
Installation Costs		
Charger	2000	$
Charge Station cost	400	$

Control Unit	1500	$
Transportation	10%	
Factory Acceptance Test		
AGV system training		
Project documentation		
Running Cost		
Yearly cost extras	6000	$
Number of AGVs	2	
Years Active	1	
Total AGV Cost years Active	$387,056	

Using the cost table rough calculations can be made by changing the number of AGVs and active years, as an example 2 AGV s operating for 1 year will cost approximately $380,000+ when using automated forklifts. However, the following years costs will be approximately $87000.

6.4 PERFORMANCE CRITERIA

There are many ways to assess an AGV fleet performance. Some criteria that are often used are material handling cost, travel distance, queue length (Beamon 1998). In manufacturing, the majority of performance measures are time-based. (Qi et al. 2018) states that the performance of an AGV is affected by transport efficiencies, such as traffic-control policies and warehouse layouts. The paper then focuses on algorithms that mitigate collision and solve deadlocks. When trying to mitigate deadlocks and collision, it is key to have a flow path design that can achieve the required performance criteria.

A common method in performance optimisation is the minimisation of the total vehicle travel distance from a given layout (GASKINS & TANCHOCO 1987). Further study by (Kaspi, Kesselman & Tanchoco 2002) factors in the travel time with a loaded and unloaded vehicle. An objective function used by (Lim et al. 2002) further progress as it uses travel time which includes the loaded and empty vehicle travel whilst also factoring in the waiting time caused by congestion or vehicle interferences. Based on this knowledge, the algorithm does have the potential to be further increased by incorporating the algorithm discussed earlier in (‘Improving safety and efficiency of AGVs at warehouse black spots - IEEE Conference Publication’ n.d.) where the AGV can have external sensors located at blind spots and junctions that can allow for AGVs to progress without having to slow down if the area is deemed safe to do so.

In a paper by (KIM & TANCHOCOJ 1993), the travel cost is a major factor as well as the cost of each path segment. In another paper by (Chen, Mcginnis & Zhou 1999), a model is developed from which the total vehicle travel time, as well as the trip failure rate, is used as a performance measure. (Talbot 2003) factors in the number of AGVs that are required and the guided path length to measure the overall performance of the system. It can be seen that overall using multi-criteria objective functions. A better-quality outcome can be achieved.

Overall, the above papers achieved satisfactory results with each method. However, it can be seen that there is a clear focus on cost, travel time of the AGVs and the waiting time in the AGV queue. In a paper by (Johnson 2001), a focus on the empty vehicle traffic is highlighted, and it shows how strongly this traffic influences various aspects such as flow path design and vehicle requirements. This is because an empty vehicle holds a greater cost as it takes up space on the warehouse floor as well as energy and travel time costs.

6.5 AGV SCHEDULING AND DISPATCHING

AGV scheduling and dispatching are the most crucial aspects of an AGV system as it is responsible for when, where and how an AGV is to operate. An AGV scheduling system normally consists of 2 types of schedule. This is offline and online scheduling. Offline scheduling is when you know in advance what the AGV is to-do and, therefore, can have everything already pre-planned. Online scheduling is for a more dynamic environment. Offline scheduling is the least favoured due to the fact that if there is a change, it can result in the failure of the system (Le-Anh & de Koster 2004). Thus, online scheduling is the most widely used method. However, when integrating a dynamic system, it must be noted that one of the most crucial things, which has previously been mentioned, is to prevent deadlock while selecting routing (‘Conflict-free shortest-time bidirectional AGV routeing: International

Journal of Production Research: Vol 29, No 12' n.d.). For this reason and in order to save time warehouses have started to integrate multi-load AGVs. This is where an AGV is allowed to detour from the shortest path as long as it is picking up something on the way (Tanchoco & Co 1994). In the paper by (Bilge & Tanchoco 1997), the benefits of integrating multi-load AGVs are that the system becomes significantly less bottlenecked.

6.6 KNOWLEDGE GAP

The above literature review holds valuable information that will allow for a good foundation when designing the AGV fleet system. However, there are some key areas that have been identified as a knowledge gap and are described below:

- The research area of AGV integration seems to mainly focus on optimal paths instead of the optimal number of AGVs.
- A key area that is missed is the overall reliability of an AGV system and longevity, i.e., when will an AGV need to be replaced due to longevity or upgrade.
- Development of adaptable AGV Systems for fleet management

For the purpose of the thesis the key aspects that will be focused on is the development of an adaptable system where the number of AGVs can be changed and the effects on a system can be viewed.

7 METHOD AND RESULTS

A method is a crucial aspect when it comes to an understanding the project and developing it. In order to solve the problem at hand, a certain procedure must be followed in order to achieve satisfactory results. The flowing diagram displays a methodology that can be followed for simulation software and results verification.

Removed due to copyright restriction

Figure 7 Software methodology diagram

Looking at the diagram, it can be seen that there are three main stages in the methodology. These stages are Problem Understanding, Software Research and Simulation Process, which in return will produce the results of the system.

7.1 PROBLEM UNDERSTANDING

In order to achieve the required results, a good foundational understanding is needed about the problem. This involved having numerous meetings with key people at Flinders and the defence company in order to produce the required outcomes of the system. The key outcomes from these meetings are discussed below.

The current method of AGV calculation:

Currently, there is no method of calculating the number of AGVs needed. This is due to the shipyard being under construction and zones not being finalised. Thus, having a tool that is adaptable is ideal for the defence company's current situation.

Visualisation of the system:

The defence company would like a system that is visual as a non-visual system could lead to major problems such as incorrect data input, thus giving skewed results or a false final answer. Without the visual system, this type of error could be hard to detect. The visual system will at least display the system such that a view would be able to have a better understanding as to what went wrong. The visual system will also allow for better communication within the various departments or among businesses when trying to convey the reasoning behind their decisions for the number of AGVs required.

7.2 TESTING SCENARIOS

It is understood that it is not truly possible to test the real time behaviour of a system until it is built and placed into use within the shipyard. This has been taken into consideration from the defence company, however, having the AGV system will allow them to improve their designs without adding extra cost or getting an external company to keep readjusting quotes.

7.3 SOFTWARE REQUIREMENTS

When deciding on the software to use in a project, there are four key steps that need to be implemented in order to choose the best software for the job.

The first step is to look at the big picture and create a simple plan. This step is important as it takes into consideration key aspects such as who the people are that will be using the end software, the cost available for purchasing the software as well as the timeline, thus factoring in the overall usability of the software for the developer.

The second step is prioritising the needs of both the end customer, which in this case is the defence company and the programmer. There is never a one size fits all when it comes to software development as every software has trade-offs. Thus, the following aspects that this step considers when choosing the software is:

- Must – These are the features that the software must include in order for it to be a viable option
- Should – The features that the software should have but are not necessarily a deal breaker if the feature does not exist
- Could – These features will be nice to have in the software

Some of the other key criteria notes that should be considered:

- How well supported is the software?
- How many tutorials are there to learn the software?
- How reputable is the company?

The defence company wants to incorporate AGV simulation software to gain better results prior to integration, know the behaviour of the system, solve any problems they may have during the designing process and obtain the optimal number of AGVs in various scenarios. There is a wide range of software's that can be used to create this simulation, with varying tools and functionalities. Thus, in order to determine the software of use, research varying software's were researched.

In order to determine the software to be used, a method called a MoSCoW is used. This method was developed by Dai Clegg in 1994. It works by looking at a series of features based on importance. MoSCoW stands for Must, Should, Could and Won't. for this stage of the process. The won't is used as this aspect is hard to label due to the nature of the software being investigated being very adaptable since it's built from scratch. Must is the most crucial parts that the software must satisfy in order to be useful for both the developer and end customer. Should are features that are helpful for the development of the software includes these and could are features that do not need to be present but would be nice to have.

Table 2 MoSCoW on the requirements for the software to build the system

	Must	Should	Could
Easy to use			
Prior Knowledge			
Tutorials for software			
Adaptability			
Visualisation			
Free version			

Using the MoSCoW diagram above, three pieces of software were finalised upon to be Unity, Gazebo and MATLAB. In order to make the final decision on which software is used, a Pugh matrix is used. The Pugh matrix works by assigning a positive score when the software satisfies the given criteria or a null score if it does not. The final weighting shows which software should be used as it covers the greatest aspects of features.

To make the final decision about which software fulfils the company´s requirement, the Pugh matrix method has been applied. This technique, designed by Professor Stuart Pugh, evaluates several design concepts through different criteria (Burge, 2009). If a concept design satisfies the criteria, it receives a positive score; otherwise, it receives a negative score or null score. These criteria can also be weighed with a higher or lower rate regarding how important it is for the project.

Decision to be made:

	Unity	Gazebo	MATLAB
Easy to use	3	2	3
Prior Knowlege	4	3	4
Tutorials for software	5	5	5
Adaptability	4	5	4
Visulisation	5	4	3
Free version	5	5	5
Score	26	23	23

Figure 8 MoSCoW diagram for software to create the adaptive fleet model

All of the above software can be used to create an AGV fleet management system; however, the Unity program scored the highest. The key factors that set Unity out are that there was prior knowledge on how to use the software, allowing for a quicker grasp of the system. Many online tutorials for the software and the fact that it's free and designed for the visualisation of games or systems. Using Unity provides the necessary freedom to experiment and adapt the model in a more user-friendly way than the other two as well as giving the ability to easily compile the project into an application that can be run independently on other computers. Unity does allow the option to integrate ROS into it, allowing for some greater control in robotic systems; however, it is more ideal to use the inbuilt unity functions with C# due to the learning that will still be required in completing the project and the available online resources.

7.4 OPTIONS

Options for attempting to create an adaptive AGV fleet system

When creating an adaptive AGV fleet, many different approaches can be taken to create a successful simulation. In order to evaluate the specific options to see which is most viable to attempt, Table 4 below is created, with the options as given.

Option A: Use industry software to create a large scale system of the shipyard

In this option, the current shipyard can be modelled inside current industry software such as AnyLogic to provide a realistic model of how the shipyard might work. However, this option does not allow for full customise ability and has a more significant learning curve. Another problem that is encountered with this is that to make a system at that scale. The software will need to be purchased to allow for both the complexity in the number of comments and allow for time frames of greater than 1 hour. Thus, it is not an ideal solution due to the costs needed to create the system.

Option B: Use Unity with a large scale shipyard to create the system

With a project like this, one option in creating an adaptive shipyard AGV fleet is to use the current existing shipyard as the base and design it such that the AGVs can traverse to different warehouses on the base. This method's pros are that it allows a full-scale model and real-time estimations as the AGVS will likely be traversing the same types of distances between the various buildings.

Option C: Use Unity to create a smaller system to represent the context of the shipyard

This option will allow for a more straightforward proof of concept showing the logic of the system rather than the entire system in action in the shipyard. The pros of this system are that it will allow for faster testing of the software and the code allowing for real-time speeds to be tested and faster simulation speeds. This method will use C# in Unity.

Option D: Use industry software to create small scale system

This option uses the industry software AnyLogic to create a proof of concept of the AGV system on a small scale simulated to option C.

Using the evaluation scale in Table 3 below the 4 options are compared to each other in Table 4

Table 3 Options Evaluation Scale

Evaluation Scale		
1	**2**	**3**
Bad	**Fair**	**Good**

Table 4 Evaluation of Options

	Weighted Value	Option A	Option B	Option C	Option D
Visual representation of the system	6	2	3	2	3
Difficulty	9	2	2	2	2
Time to implement	3	1	2	3	1
Flexibility	9	2	3	3	2
	Total	**51**	**69**	**66**	**57**

Looking at Table 4 above it can be seen that the best option for this project is B where the current the defence company shipyard is used to create the test simulation.

7.5 OPTION B

Create a 3-Dimensional (3D) model of the shipyard:

For the 3D model of the shipyard, a program called SketchUp was used in conjunction with the site plan. Sketchup is a 3D modelling software that is primarily used in architectural, interior design, landscape architecture, civil and mechanical engineering. In order to create a working model for the system, the buildings, floor and path are created separately. By doing this, Unity is able to identify each set of objects as their own entity, thus allowing for specific scripts to reference the CAD files independently.

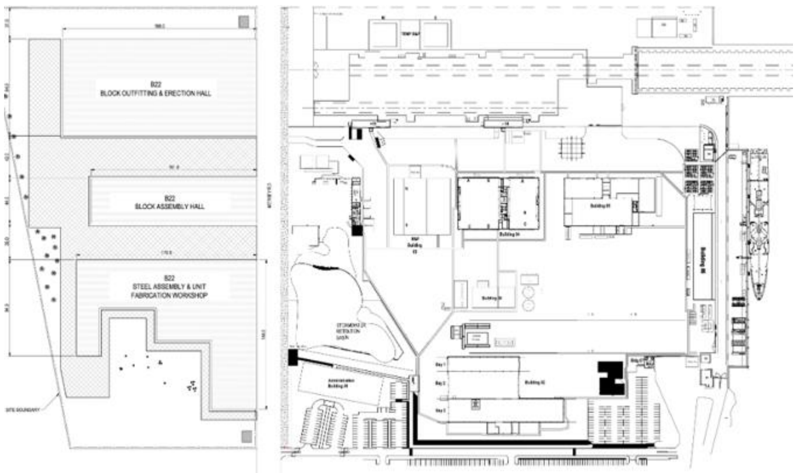

Figure 9 Overview of the shipyard with the new buildings on the left hand side

The Cad files are imported into unity and are scaled 200:1. This reduces the overall polygon count of the simulation, allowing it to run smoother. This was done as it does not have any impact on the outcomes of the system.

Create delivery points:

The initial simulation involved creating nine drop-off locations, and three pickup locations and an AGV spawn location. The spawn location was an arbitrary location that was chosen and can be easily adjusted. Figure 10 shows the location of the various points.

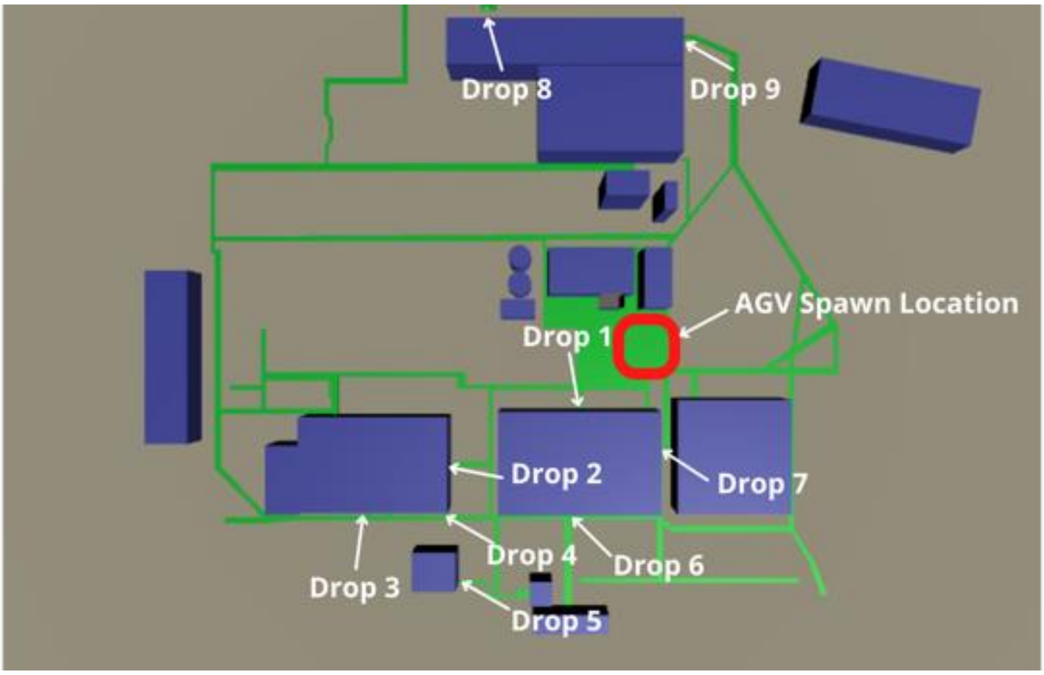

Figure 10 location of the various pickup and drop off points for the shipyard, green is where the AGVs can travel

AGV testing model:

In order to create a simulation model, the AGV was set as a sphere. This added a layer of simplicity and did not obstruct the functionality of the AGV controller. The MIR robot was a potential model to use; however, these are not outdoor AGVs and could change the user's perspective on the system as they might focus on the functionality of the specific robot instead of the system as a whole which is the aim of the project.

The model functions in a very linear fashion where the user initiates the number of AGVs and number of resources, then spawns the AGVs and allocates the AGVs to a specific pick up point when it becomes available. Since the location of the resource is known, the AGV uses A* navigation to calculate the fastest route to the location. Once the AGV reaches the pickup location, it is then allocated a drop-off point at random, and the pickup location is freed up for another AGV to pick up resources. Once the AGV reaches the drop-off point, it is then assigned a new pickup location, and the cycle continues. The diagram below gives a clear indication of the process.

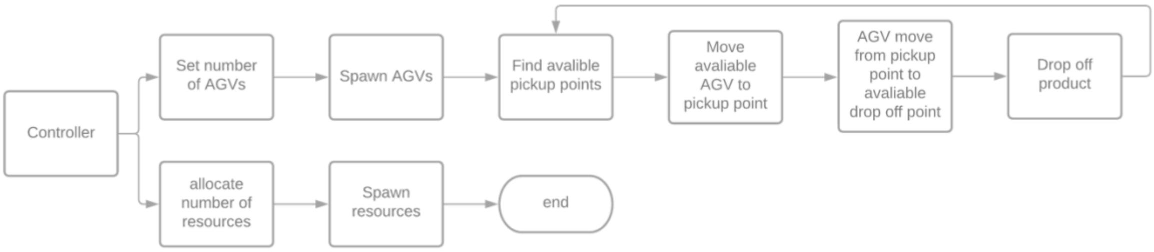

Figure 11 Option B block diagram of the main AGV controller

- Set Number of AGVs
  - For this model, the number of AGVs is set at the start of the simulation. This is done by taking the user input prior to starting the simulation. The number is then grabbed by the AGV spawn object.
- Spawn AGVs
  - The spawn object uses the AGV object to instantiate the required number of AGVs that the user set.
  - AGV object: this object is designed to take input from the AGV controller where it can be given specific tasks mainly focusing on objectives such as given a location to traverse to. The object uses A* navigation to manoeuvre to the given location.
- Find Available pickup location.
  - The AGV Controller is given an array of the pickup locations. The number of locations can be any given number; however, the locations must be predetermined and allocated a game object. By allocating a game object as the location, the AGV controller can see it. The AGV controller will select a location from the array and lock it, preventing other AGVs from being allocated to this location.
- Move available AGV to pick up point.
  - Once the AGV controller has been allocated a location, it will continuously move towards the location using A* navigation until it is within a suitable range of the pickup point. At this point, the AGV will notify the controller that it has

arrived. The controller will then, at random, designate a drop off location from the pre-specified array of the drop off locations.

- Move AGV to drop off point.
  - Using the method discussed above, the AGV will navigate to the designated drop off point.
- Drop off product
  - Once at the drop-off point, the AGV will then allocate itself as free and notify the AGV controller.
- Allocate resources
  - The resources for this simulation have been given as pipe spools, bolts, and nails. These are just arbitrary things; however, based on the current setup provide no significance as the system does not allocate based on resource request but at random.
- Spawn resources
  - There are fields that allow for the user to input the required number of each of the three materials; however, these fields go unused as the AGVs are randomly allocated to move components to the various drop off points.
- Resource request
  - Resource request works by randomly generating a certain amount of resources between 0 and 10. The resources are then linked to each of the location objects that have been pre-selected before the simulation starts.

7.5.1 OPTION B RESULTS:

To test the viability of the simulation, 3 tests were conducted on the AGV system. Each test consists of 3 runs allowing for a total of 9 simulations to be run and evaluated. Since the model did not have any internal timing systems, the AGVs were timed with an external timer. The AGVs were manually sped up in order to save time. The simulations are such that when all the AGVs reached a deadlock, the simulation would be considered done, and the timers stopped.

The number of AGVs 3:

Looking at the table and figures below in table 3 and figures 12 and 13, it can be seen that the AGV simulation did not last very long until deadlock occurred. As can be seen, the system outputted an actual Idle time between 28 and 32 percent of the time. This is significantly larger than the 0 percent it should be. The reason the system should be at 0 percent Idle is that in the simulation, there are three pickup locations that exist and nine drop-off locations. This means that the AGVs should not have any downtime as they should be constantly active with either delivering or picking up a package.

Table 5 Runtime until deadlock 3 AGVs

Run	Idle	Delivering	Pick up	Idle % (Should)	Idle % (Actual)
1	2:35	2:52	2:41	0	32
2	2:05	2:31	2:07	0	31
3	1:55	2:30	2:33	0	28

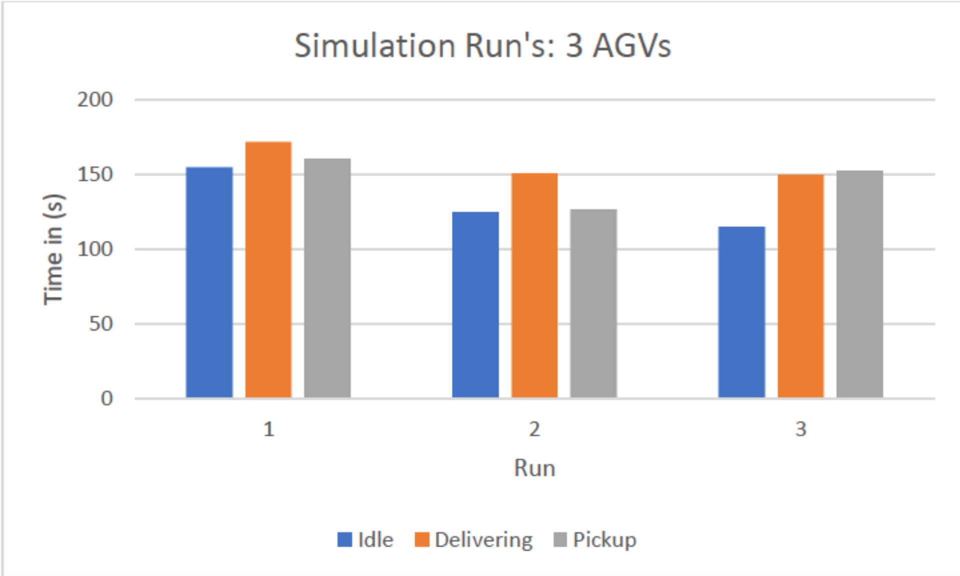

Figure 12 Idle, delivery and pickup times until deadlock occurred for 3 AGVs

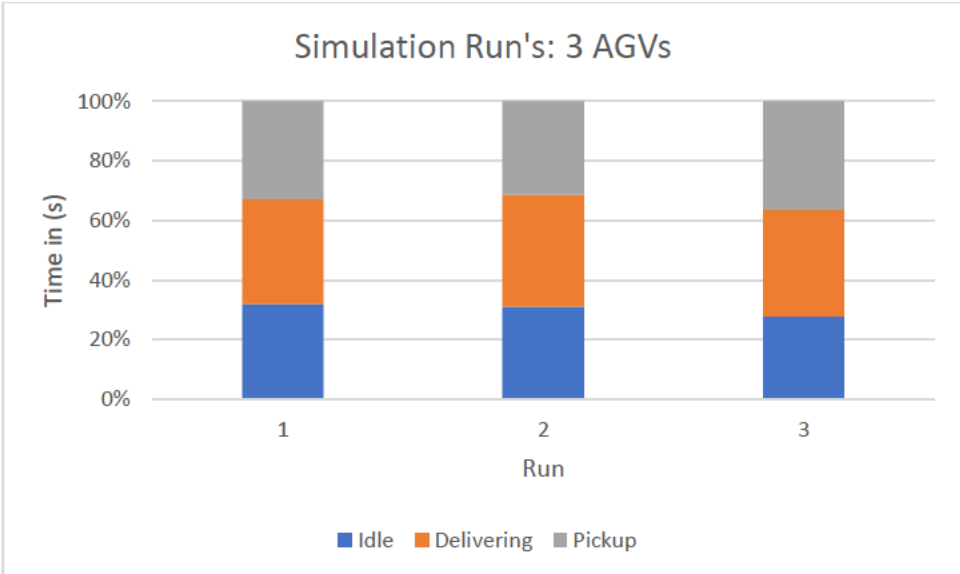

Figure 13 Idle, delivery and pickup times until deadlock occurred in terms of time percentages for 3 AGVs

When looking at the three simulations, it can be seen that there is only a slight variation in the AGV values between the three simulations. It can be seen that the first run took the longest amount of time; however, in the time was spread evenly between idle, delivering and pickup.

Looking at the results of the 3 AGVs, it is clear that there is a logic error as the AGVs are getting locked up and once at a delivery location are not able to move for a random amount of time from which the AGV will not get locked up and begin to move. The reason for this is because the system works in a linear fashion from which every frame checks an if statement to see if there are any AGVs ready to use, then a separate one to see the other AGVs that are in use. Unity runs the code using a constant update loop, which directly associates with every frame of the program, allowing for real time visualisation of the system. However, because it's checking an array and only using the first element in the array, the program will often skip over and erase other AGVs in the array. This causes the AGVs to not get assigned tasks causing the deadlock.

The number of AGVs 6:

Due to the initial start time of the AGV system and the fact that only 3 AGVs can be assigned at a time to pick up, and Idle value of 10% is estimated. The data in Table 6, Figure 14 and Figure 15 shows that the AGV had Idle time between 31 and 34 percent.

Table 6 Runtime until deadlock 6 AGVs

Run	Idle	Delivering	pickup	Idle % (Should)	Idle % (Actual)
1	3:42	4:09	4:15	10	31
2	296	276	287	10	34
3	4:10	4:29	4:46	10	31

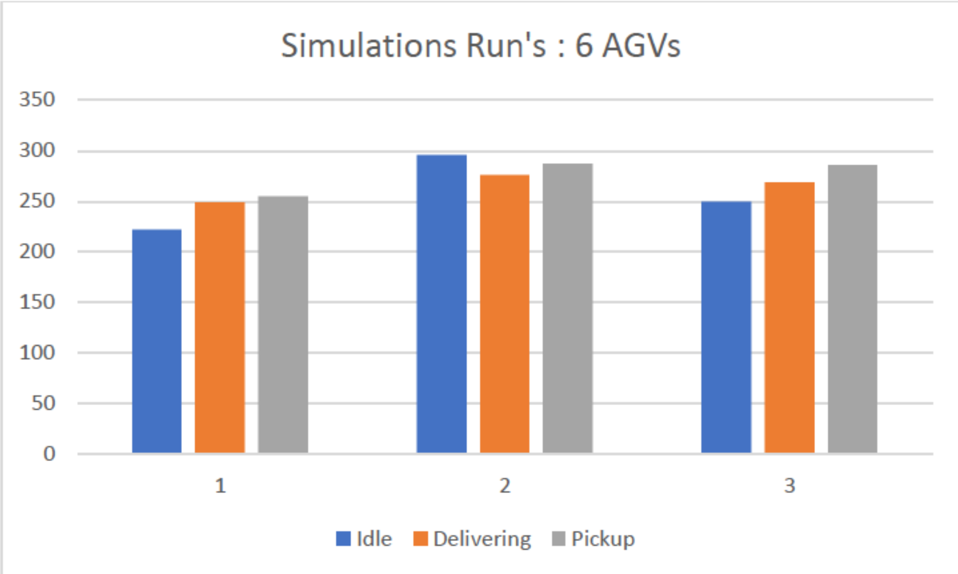

Figure 14 Idle, delivery and pickup times until deadlock occurred for 6 AGVs

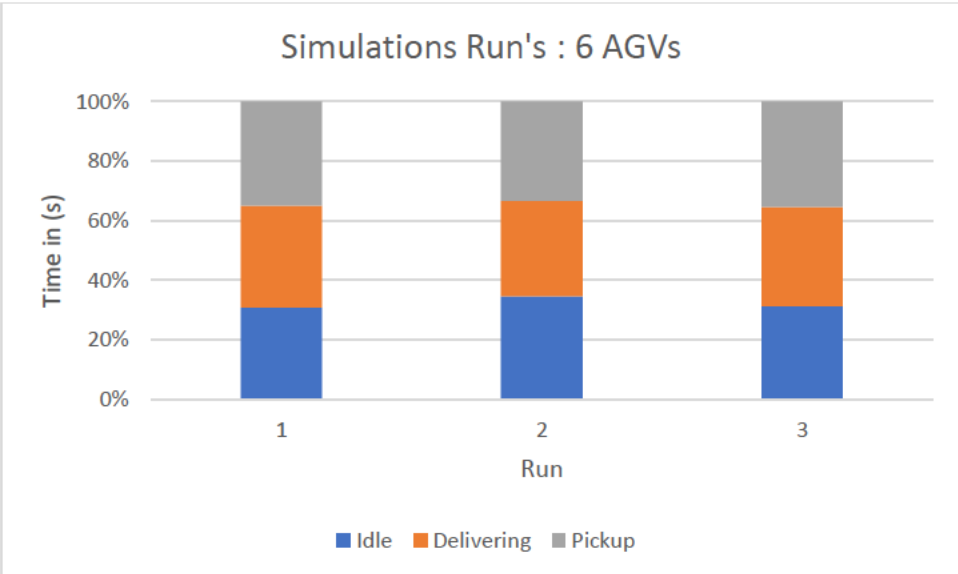

Figure 15 Idle, delivery and pickup times until deadlock occurred in terms of time percentages for 6 AGVs

The simulation behaves more realistically than the previous simulation in terms of idle actual vs what the idle value should have been. However, the actual idle value is almost identical to that of the 3 AGV run. This value is definitely not correct, as doubling the AGV number should have a large impact on a normal system, further proving the logic fault of the system.

The number of AGVs 12:

Due to the initial start time of the AGV system and the fact that only 3 AGVs can be assigned at a time to pick up, and Idle value of 25% is estimated. The data in Table 5 and Figure 16 and Figure 17 shows that the AGV had Idle time between 33 and 35 percent.

Table 7 Runtime until deadlock 12 AGVs

Run	Idle	Delivering	Pickup	Idle % (Should)	Idle % (Actual)
1	4:00	3:56	3:59	25	34
2	4:56	4:21	4:56	25	35
3	3:59	3:54	4:01	25	33

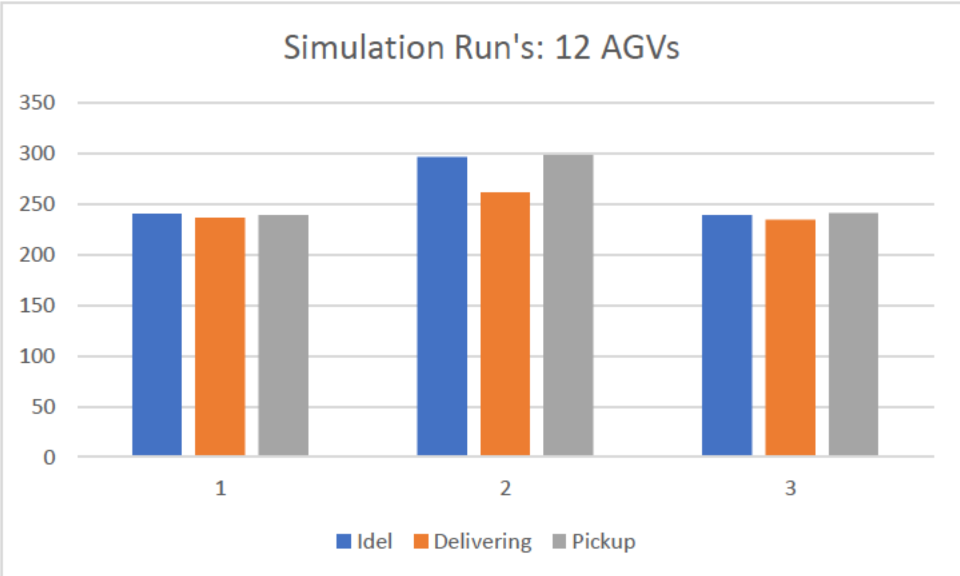

Figure 16 Idle, delivery and pickup times until deadlock occurred for 12 AGVs

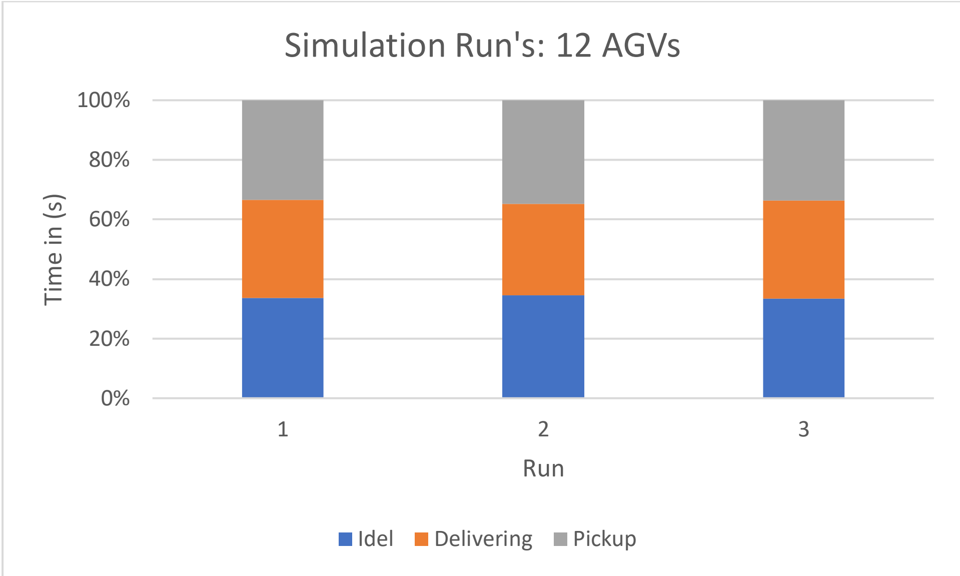

Figure 17 Idle, delivery and pickup times until deadlock occurred in terms of percentage for 12 AGVs

As the number of AGVs increased, the Idle times became closer to the values that they should be; however, as discussed above, the idle times are staying very consistent as can be seen, the idle value of 12 AGVs is almost identical to when there were 3 AGVs.

Problems with simulation:

Overall, it can be seen in the Figure 18 below that changing the number of AGVs in the system had little to no effect on the system as all the measurable points of the system. This further proves that the simulation is not viable and needs major improvements. It should be noted, though, that the idle time did increase as the number of AGVs increased. This is what is expected; however, this result can be assumed to be unrelated based on all other evidence of the system not working.

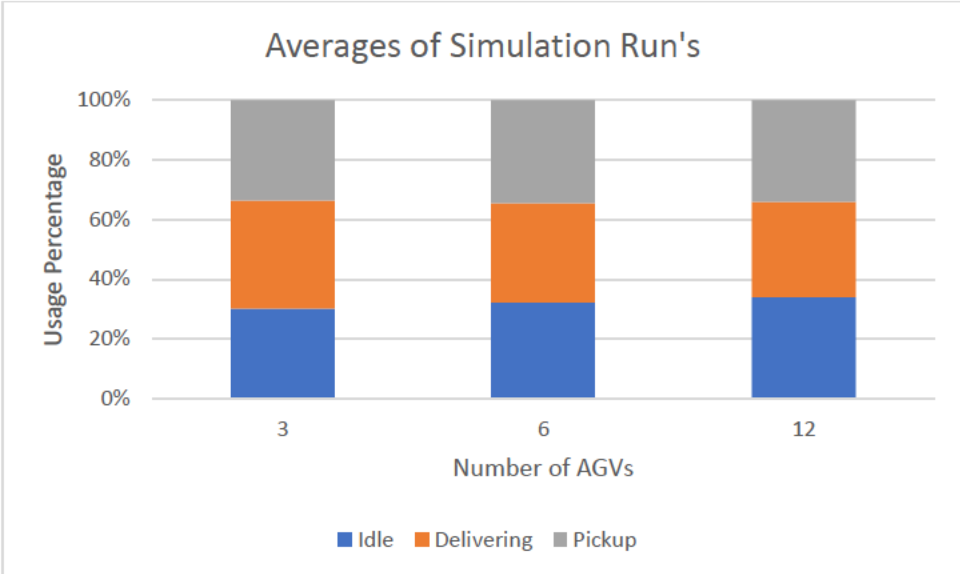

Figure 18 Comparison of time percentages between 3,6 and 12 AGVs

Implementing a system like this on a real scale caused considerable problems with the simulation preventing it from achieving any usable results. There are four key problems with the simulation discussed above in the Figure 10.

1. Flawed logic
    a. Looking at the diagram above, it seems like the process should somewhat work where essentially a AGV gets an object, drops off an object and gets another one. However, due to the scale, the system was running on some issues that arise with the logic. A major issue that can be seen in the model is the fact that once a pickup location is claimed by an AGV, it cannot be claimed by another AGV, which means that an AGV on the other side of the yard can claim a spot, and the AGV next to the pickup location will not be able to move. Shown in Figure 19

Figure 19 AGV grouping at a single dropoff location

b. This leads to the second issue with the logic. Since the AGV cannot claim a pickup location, it will wait at the drop-off point, preventing other AGVs from delivering their object. Shown in Figure 20

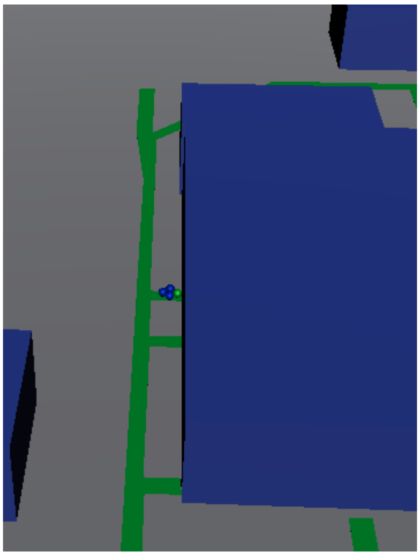

Figure 20 AGV blocking others from delivering parcels

c. Various other factors also contributed to the flawed logic as the system runs many different scripts on different objects in each frame. It is potential for AGVs to become locked in an infinite loop if something did get triggered correctly.

   d. The AGVs also did not have any type of charge logic. Thus, this model assumed an infinite battery which would skew the results if data was able to be captured.
2. Delivery points
   a. The system had too much of a focus on the various delivery points and not so much the adaptability of the delivery points. This means that the handover would be difficult for the software. This aspect also did not meet the main goal of adaptability since everything was very fixed and picked at random.
3. Difficult to automate data captured – collect data with timing.
   a. As the logic in the main AGV controller was flawed and caused lockups of AGVs, the data obtained was not useful and prevented most of the data from being captured and displayed correctly. However, the system was able to be timed until complete deadlock. The idle, delivering and pickup times were captured using manual timers.
   b. The only function aspect of the system was the colour changing of the AGV, determining if it had a product or not. This is more visual data for the user as opposed to actual metric data.
4. The system can only run in real-time.
   a. Due to the setup of the functions and speed controlling factors that the original scripts worked off of, there was no possible way of controlling the AGVs speed, thus resulting in a simulation that only ran in real-time, which is not feasible for a simulation of this type.
5. Collision of waiting objects
   a. During the simulation Runs, the collision of waiting objects was also a major flaw, as when an AGV was able to move to go and pickup another item, it would knock other AGVs, which would then cause them to move freely and settle in positions such as shown in Figure 21 below.

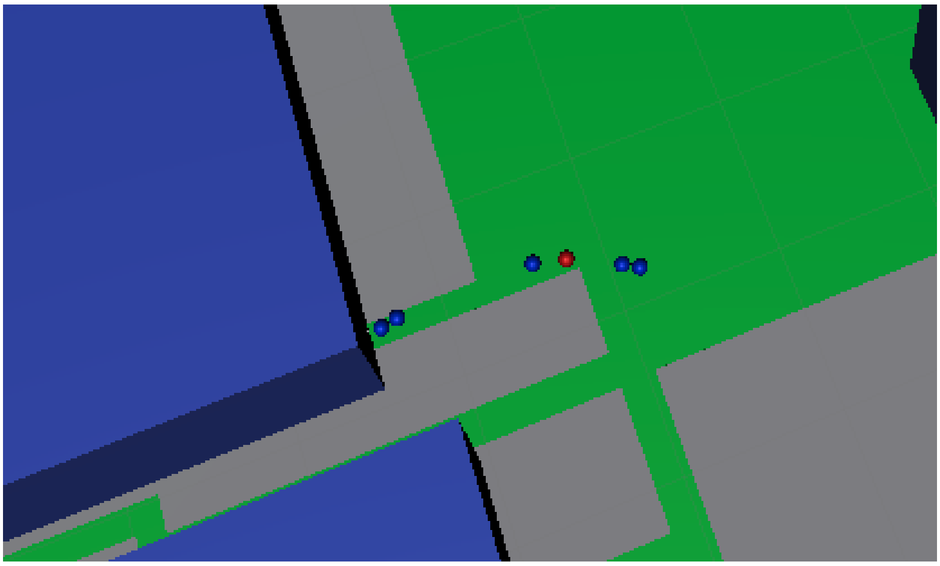

Figure 21 Collision with other AGVs causing them to roam into the middle of a path

7.6 OPTION D

Due to the shortcomings of Option B, the next option in the table suggests that option C, where a smaller simulation of the unity system is used; however, this is not ideal as the failures of option B are linked to logic errors. Due to the logic errors, it was decided to use option D, where a small test system will be created in industry software as a testing point to fine-tune the system's logic to implement it in option C then.

7.6.1 MATLAB:

Method

Due to the failure of the first model in option B another method that was attempted was to try and use MATLAB (‘Control and Simulate Multiple Warehouse Robots - MATLAB & Simulink - MathWorks Australia’ n.d.) and integrate some premade AGV scheduling systems to try and build a viable simulation. Learning from the previous mistake, the model was scaled-down, and a replica of line zero was used. LineZero is the defence company’s current industry 4.0 testing facility located at the Tonsley precinct. The benefit of using this model is the size of the area will allow for quick real-time simulation and the potential of using a real AGV to replicate the simulation.

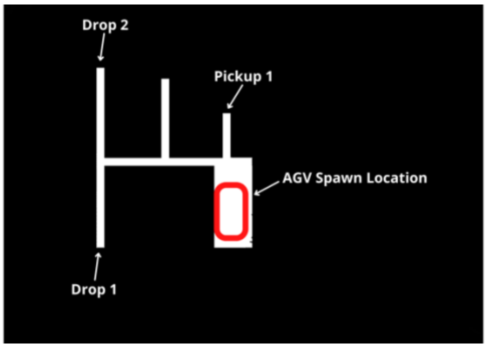

Figure 22 LineZero (on right) converted to a binary occupancy grid (on left)

7.6.1.1 HOW THE MATLAB SIMULATION WORKS

The system scheduler works by controlling the whole system output. It commands the AGVs to collect packages from the loading station and deliver them to a specific unloading station. The trajectory of the AGV is set based on the location of the various unloading and loading stations, this allows the controller to generate velocity commands for the AGV. The commands are relayed to the differential drive robot which executes the velocity commands and returns ground-truth poses of the AGV. The controller then uses the poses to track the status of the robots. Figure 23 below shows the robot control system.

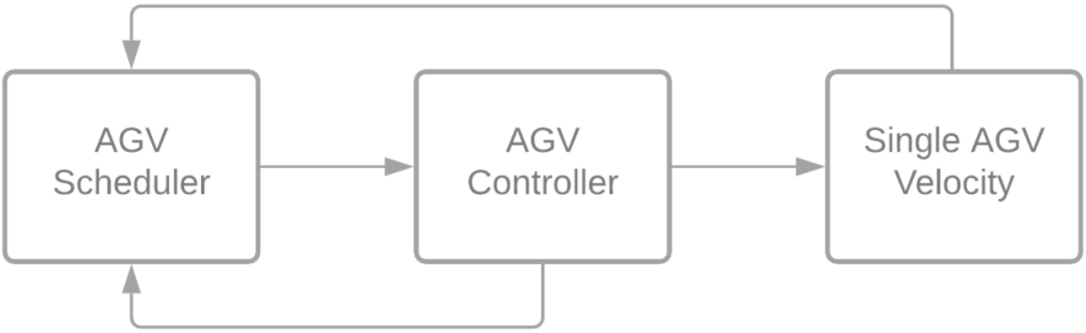

Figure 23 MATLAB AGV Control system

The AGV Scheduler uses a state controller in order to handle the allocations of the AGVs. The reason that the central schedule can work with so many AGVs at the same time is because it uses a for loop for each subsystem (Simulink) which contains an array of AGV component units that are constantly tracking each of the AGV states. This means that each iteration of the entire system involves a full iteration through all the AGVs giving the ability for all of them to look and function simultaneously as all the commands are output at the same time to the various AGVs.

The robot controller of the system gets given a delivery command that contains information the allows the robot to plan the path to the next delivery or drop off point. Once the path is determined, velocity commands are generated, allowing the robot to move towards the object. The robot uses a similar type for each loop in order to update all of its components and notify the system when it has reached its goal. Once reaching its goal, the status of the AGV is updated to allow the system.

A binary occupancy grid is made of LineZero, as shown above. This is what is used so that MATLAB can distinguish where the robot is allowed to travel. The spawn location of the AGV is then defined through a quartonian system. This can be seen in the code snippet below, where the charging stations, unloading stations and loading stations are defined.

```
chargingStations = [27,13;27,15;27,17];
unloadingStations = [12,13;12,35;21,32];
loadingStation = [29,29];
```

The simulation is then built and shown below in Figure 24. The benefits that this MATLAB simulation achieved, that phase one did not, is the ability to capture various pieces of data. This is due to the way that MATLAB is structured, allowing for data output to be easily captured and stored.

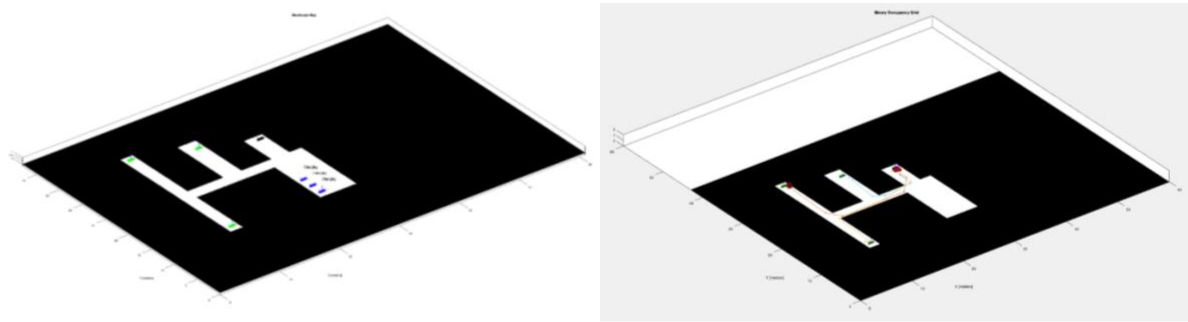

Figure 24 MATLAB LineZero AGV Spawn location (left) and parcel movement (right)

In order to test the simulation and gain an understanding of the impact that is increasing the number of AGVs has on the system, the various test was run. The test consisted of 3 runs with a fixed number of AGVs with ten randomly generated delivery points to each of the drop-off locations. The test was carried out with 2, 3 and 4 AGVs making a total of 9 runs to simulate the delivery in line zero. The test only considers the total time to complete all the tasks. This is because the way the system works will always result in 0% idle times due to the fact that the AGVs are always assigned a given task and will just wait in a queue with the AGVs, thus not contributing to idle time. The system is also only designed to consider the general effect of increasing the number of AGVs on the system.

7.6.1.2 RESULTS FOR MATLAB

Number of AGVs 2

The first test in Table 8 and Figure 25 consists of 2 AGVs in the system where the results lie between 2234 and 2304, giving an average of 2270 seconds which is approximately 37 minutes.

Table 8 Runtime until 10 parcel deliveries 2 AGVs

Run	Total Time
1	2274
2	2304
3	2234

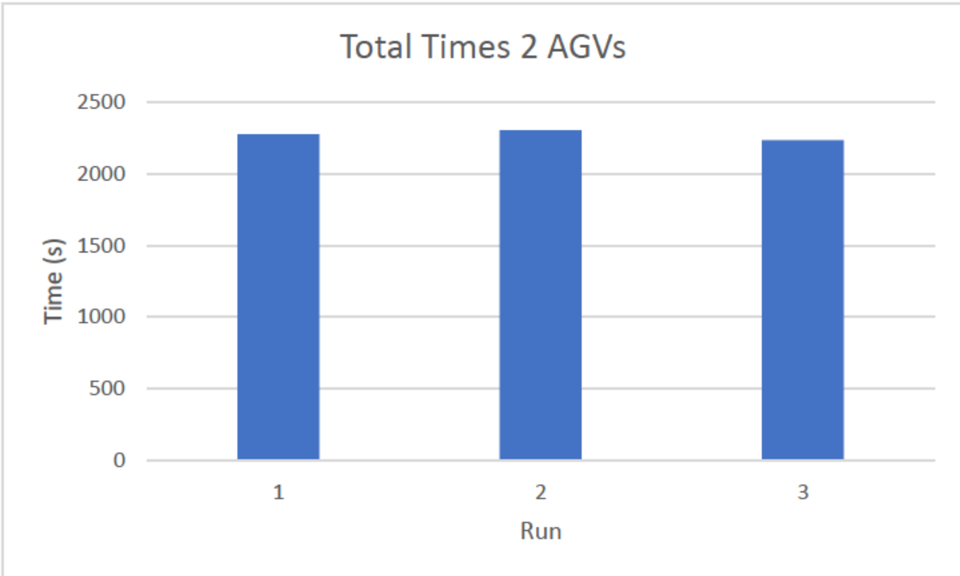

Figure 25 Runtime until 10 parcel deliveries 2 AGVs run comparison

The simulation worked as expected, where all ten packages were delivered to the stations in a reasonable time given both the distance of travel and the amount of AGVs. It can be seen that from the figures above, there was very little variability in the simulations. This means that the AGVs consistently behaved the same even with the item's delivery points being randomly generated. One of the causes for the consistency in the data is due to the fact that the pathways in the simulation are very narrow, only allowing for 1 AGV at a time. Since the systems method of collision prevention is to simply stop one of the AGVs from moving and allow the other to pass throw, it does not matter what order the delivery points are if they are evenly distributed as the AGVs will almost always perform the same, as shown above.

Number of AGVs 3

The second test in Table 9 and Figure 26 consists of 3 AGVs where the results lie between 1831 and 1845, giving an average of 1839 seconds which is approximately 31 minutes.

Table 9 Runtime until 10 parcel deliveries 3 AGVs

Run	Total Time
1	1841
2	1831
3	1845

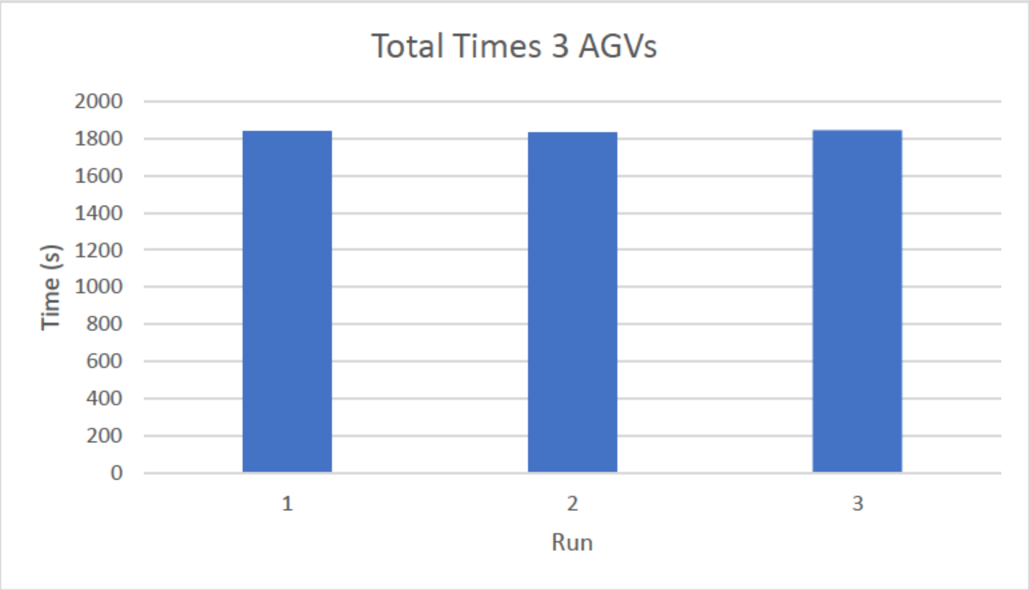

Figure 26 Runtime until 10 parcel deliveries 3 AGVs run comparison

Adding the extra AGV had an average of 19% efficiency increase on the AGV system. The system can also be seen to be behaving very similar to the 2 AGV system described above, where the outcomes of the tests were extremely similar. This is due to the same reasons as discussed above in regards to the method the system handles AGV collisions, and the distributions points are chosen at random but distributed evenly.

Number of AGVs 4

The third test in Table 10 and Figure 27 consists of 4 AGVs where the results lie between 1456 and 1599, giving an average of 1522 seconds which is approximately 25 minutes.

Table 10 Runtime until 10 parcel deliveries 4 AGVs

Run	Total Time
1	1456
2	1510
3	1599

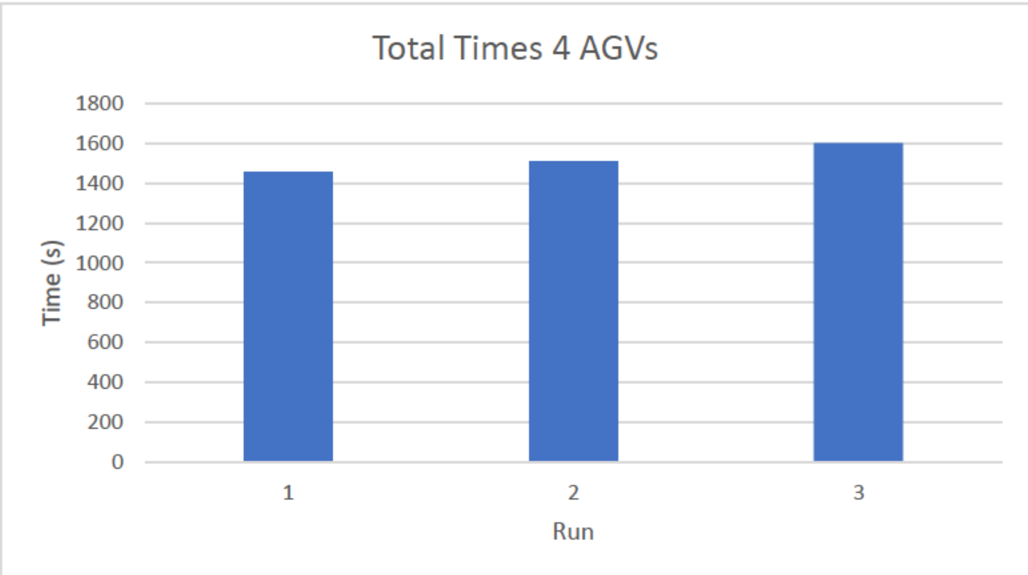

Figure 27 Runtime until 10 parcel deliveries 4 AGVs run comparison

Adding the extra AGV for a total of 4 AGVs adds 17% efficiency in the system compared to the 3 AGVs with a total of 33% increase from 2 AGVs. The 4 AGVs have a more noticeable difference between the runs in terms of time taken compared to the previous runs with 2 and 3 AGVs. The reason for the difference is because as the system has an increase in AGVs, so does the choke points increase where the AGVs collide with each other while transporting their items.

The simulation was run until the maximum of 6 AGVs. Adding any extra AGVs caused a deadlock in the system. It was found that the system could only achieve up to 6 AGVs. The Figure 28 below can be seen containing the results merged with the previous simulations.

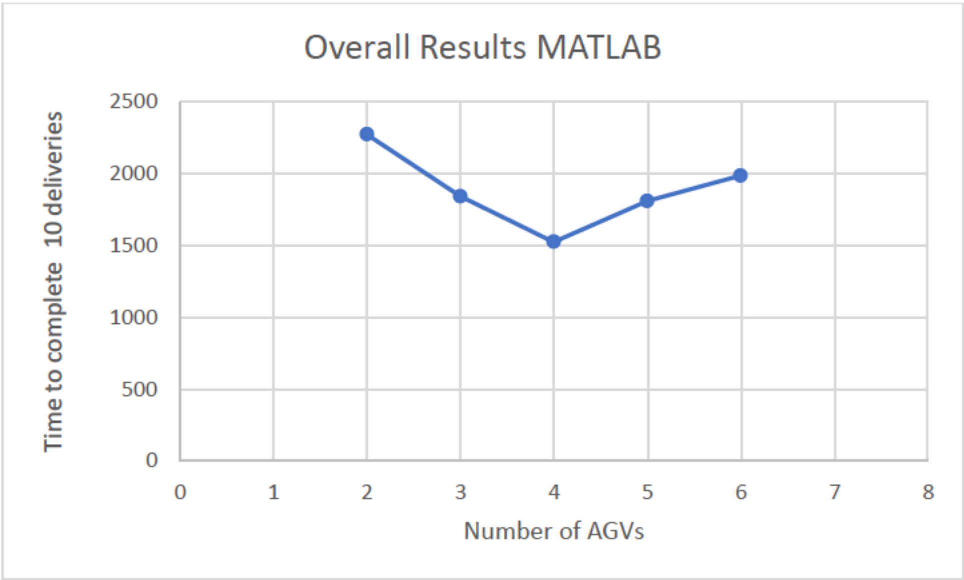

Figure 28 Runtime to number of AGV ratio for all runs

Looking at the simulation in Figure 28 above, it can be seen that the AGVs follow a linear line of improvement up until 4 AGVs. After 4 AGVs, in this case, most systems would start to plateau as increasing the number of AGVs has no further impact on the system. In the Figure 28 above, the system starts to increase in the time values instead of plateau. This shows that after a system with 4 AGVs, adding more starts to cause a significant amount of congestion, which also justifies why the system could not handle more than 6 AGVs as the congestion becomes too great for the AGVs to navigate. However, it should be noted that the system only went into deadlock on 7 AGVs and 8 AGVs on their last two parcel delivery. The reason for this is likely due to the AGVs idling around the loading point instead of heading back to base, thus further justifying the importance of moving AGVs away from the system when they are not in use.

Another observation that what witnessed in the results is the fact that as the number of AGVs increased, so did the variability in the Range of the times for the system. The reason for this is because since the system randomly generates its delivery points for the items, the AGVs will encounter different choke points, and the more AGVs that get added to a system, the greater the choke points become. There are two main methods that can be used to combat this. One being that the AGV speed is increased, allowing the AGVs to traverse the map faster will allow for quicker movement through a choke point. The second method is to add more space for the AGVs to travel. This could either be with a wider area for which the AGVs can move through or by controlling the paths for which AGVs can travel. The major choke point for the simulation can be seen below in Figure 29

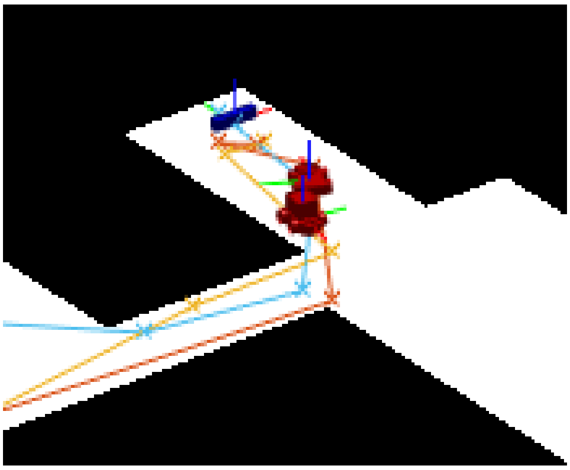

Figure 29 AGVs Moving through each other

Another issue that skewed the results was the fact that the simulation allowed in odd circumstances for the AGV to travel through the walls of the building, as shown below inf Figure 30

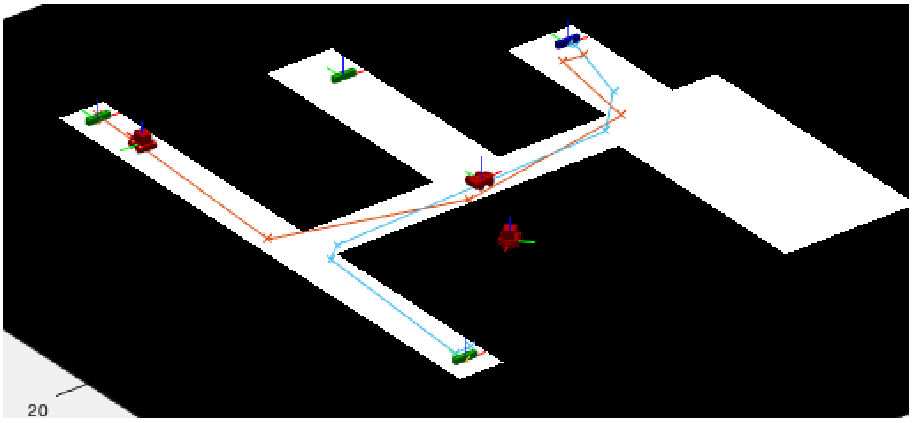

Figure 30 AGV moving through walls

This problem was caused due to the close proximity that AGVs could get to the wall, and once crossed, the occupancy grid broke the logic of the AGV as to where it is allowed to travel. The simulation setup is also difficult as the occupancy grid needs to be created from an image that is built from solid whites and blacks. When trying to create the shipyard on the size needed to simulate the AGVs accurately will be cumbersome, as shown below. The AGV spawn, pick up and drop off locations will also need to be exactly placed using the quaternion coordinates. This then reduces the adaptability for the defence company as the shipyard changes. Running varying simulations will too be difficult.

Figure 31 Failed attempt at creating binary occupancy grid for shipyard

7.6.2 ANYLOGIC:

Taking the knowledge gained from the past two attempts, it was decided that the best approach would be to use software that is used in industry to simulate AGVs in order to gain an understanding of how these systems work and how the software handles the AGV logic and fleet management. The software, as previously done, was finalised down to 4 main pieces of software by using a very similar MoSCoW diagram; however, this time, the prior knowledge is not as important as the software need to be learnt due to there being no prior knowledge.

Table 11 MoSCoW for software criteria

	Must	Should	Could
Easy to use			
Prior Knowledge			
Tutorials for software			
Adaptability			
Visualisation			
Free version			

The final decision was then again decided with a Pugh matrix where the final software was found to be AnyLogic, FlexSim, PlantSimualtion and Arena shown in Figure 32.

Decision to be made:

	Anylogic	FlexSim	Plant Simulation	Arena
Easy to use	5	4	4	4
Tutorials for software	5	5	3	3
AGV fleet size calculation	5	4	4	4
Detailed output data	5	4	5	4
Free version	3	5	4	4
Score	23	22	20	19

Figure 32 MoSCoW diagram for industry software to learn how to create an adaptive model

All of the software above has the ability to be used to simulate the AGVs in the shipyard; however, AnyLogic received the highest score. This is due to its user-friendliness of the program. There are various tutorials the software thoroughly goes through in step by step basics method. This is ideal, especially due to the time frame that the software needs to be learnt in. A further bonus is that AnyLogic provides students with full access to the software to use all the features, with the only downside being that the simulation can only go for one hour. This does not affect the project as the software is mainly used to learn how an AGV fleet is controlled in industry.

After having a further meeting with the defence company, a simple system area was decided to be used for testing. The system is one distribution centre and three warehouses shown in Figure 33. The simulation works by having the number of AGVs and their travel speed selected at the start of the simulation. Goods will then come into the warehouse, are stored, and then moved to the various warehouses. The AGVs, in this case, are separated into indoor and outdoor AGVs.

The unique aspect that AnyLogic offers is the fact that the view can be switched during simulation to 3D or 2D statistical. The 2D statistical is set up to track the AGVs travel time to and from delivery while also tracking the Idle time of the AGVs. The view also provides a heat map of the AGV travel locations allowing for trend lines in the AGVs path to be identified.

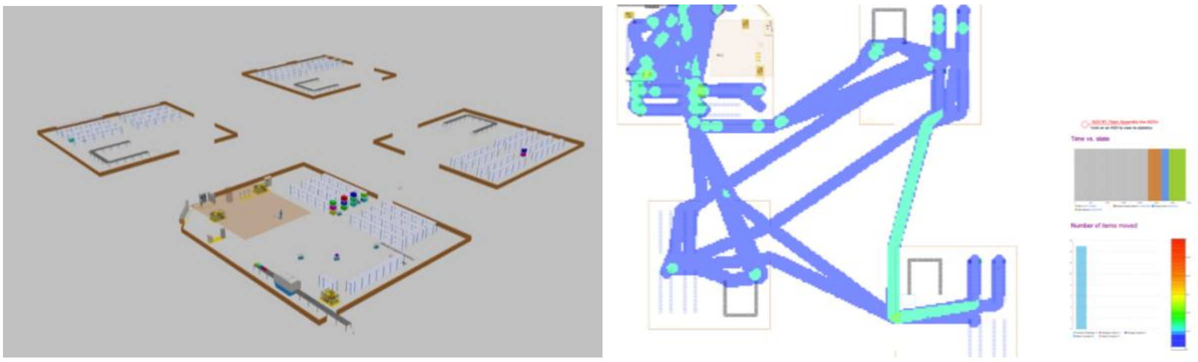

Figure 33 simple warehouse distribution simulation

The key learning that was achieved by creating the simulation in industry software is the method at which the system controls the AGV. Since AnyLogic used block programming, the logic can easily be broken down. This then allows for a strong foundational understanding of the process in which AGVs are allocated loads and how the AGV interacts with the loads and loading area. This new gained logic is the foundation for the final Unity model in an attempt to correct the first failed project. The new logic can be seen in Figure 34

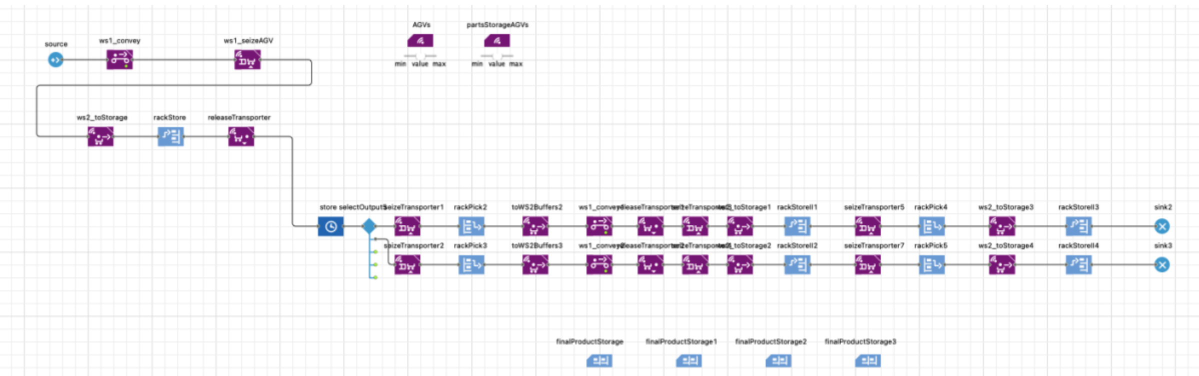

Figure 34 Logic diagram created to move parcels with an AGV fleet

The logic Flow:

1. Parts to be delivered come in

The start of the simulation involves setting the incoming goods to 2000. This is done as the student edition of the software only allows for 1-hour simulation, so an arbitrarily large number was used such that the AGVs would always have items to deliver.

2. Conveyor belt moves a component.

In order to create a sufficient pickup point for the AGVs, a Conveyor belt is used to deliver 1 item at a time; thus, in order for the AGV to recognise the location, the Conveyor belt must have both a start and an end to the system detects the Conveyor belt and moves it when there are objects spawned onto it.

3. AGV Controller allocates an AGV

Once a parcel is in the loading zone, the system then allocates an available AGV. This is handled by AnyLogic built in the AGV controller. The AnyLogic AGV controller is the brains behind the system and allows for a multitude of actions to occur when using either groups or a single AGV.

4. AGV Controller selects a rack location

Once the AGV is designated a parcel, the system is able to allocate the AGV a specific rack position. For this simulation, the AGV is only given the column of where the part should be stored; it can then determine on which of the three levels to store the object depending on occupancy.

5. AGV Controller Releases AGV

The AGV controller monitors the AGVs states, and once a parcel is delivered, it triggers a release function which puts the AGV back into the system to be used.

6. AGV Controller selects AGV to move an item from the main warehouse to either warehouse one or warehouse 2

The system that was designed moves goods from the main distribution warehouses to 3 three other warehouses, the flow of materials was a number that was approximated during meetings with the defence company for test purposes where 70% would go to warehouse 1 and 30% to warehouse 2. The AGVs would then, depending on the number of deliveries occurring, move components from either warehouse 1 or 2 into warehouse 3. This process essentially simulates a small setup where components move between different stages in manufacturing to meet at a final assembly point.

7.6.2.1 RESULTS

As specified above in the MATLAB portion of the AGV system, one method for controlling chokepoints is to enable speed control. In order to test this, the AnyLogic system was set up such that it allowed for the user to input the number of AGVs and the speed at which they were able to traverse the map.

When running the simulation, it was found that the system was deterministic. This meant that there was no point in repeating any of the experiments. The experiment was set up such that there were three levels of AGV numbers tested at three different speeds. This results in a total of 9 simulations for the system. The results for the runs can be seen below in Table 12 AnyLogic AGV tests with varying speeds and Figure 35

Table 12 AnyLogic AGV tests with varying speeds

Speed	Number of AGVS	Idle	Delivering	Pickup
0.5	4	1279	1232	953
	6	1264	1238	1003
	8	1182	1296	1009
1	4	1602	1081	838
	6	1586	1039	867
	8	1650	1090	791
2	4	1721	980	831
	6	1748	979	783
	8	1764	985	746

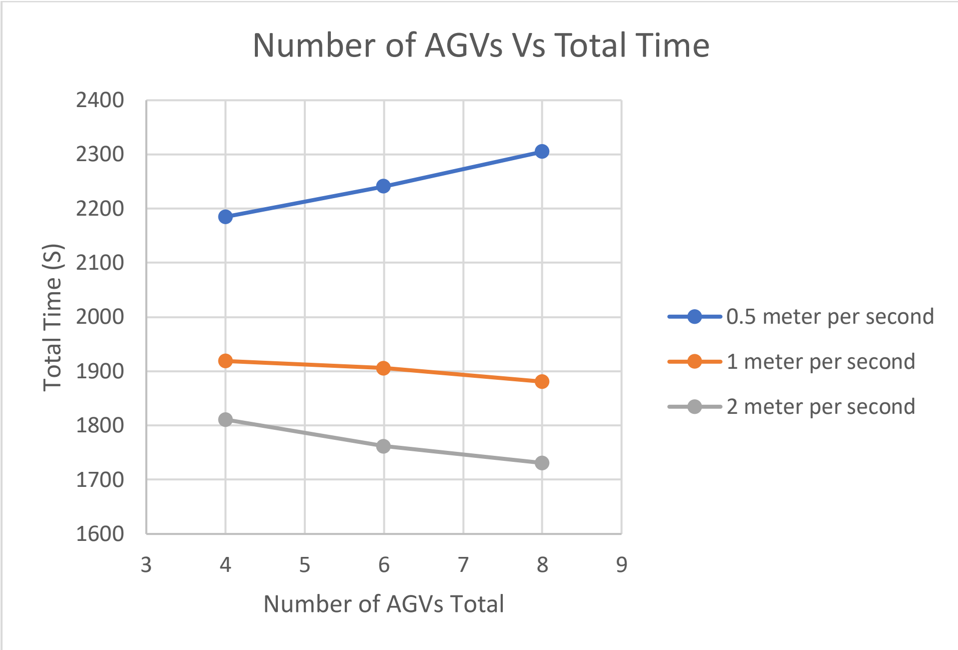

Figure 35 Amount of time spent delivering parcels Vs AGV numbers

Looking at the system, it can be seen for the most part that adding more AGVs to the system improved the overall productivity. However, in the case of the AGVs running at 0.5 m/s, it can be seen that the more AGVs that are put in the system, the worse the system got.

The reason for the increase in time for the AGVs travelling at 0.5m/s can be factored down to 2 causes. Firstly, due to the chokepoints that were encountered in the MATLAB simulation above. The second cause is because this simulation works off having 3 AGVs in the distribution centre and 1 AGV delivering between warehouses, and since the system does not track idle time due to the 1 hour of simulation time permitted, the other values of pickup and drop-off were amplified due to the slow traversal of the AGV resulting in less idle time as getting to destinations took longer.

The main chokeholds in the system can be seen in the Figure 36 below, where the green highlights are where AGVs travel and spent a longer amount of time.

Figure 36 Choke points for parcel delivery

Looking at the overall times of the system, it can be seen that the performance improved as the number of AGVs increased. This is especially prominent when the speed and the AGV increases. Figure 37 shows that when the AGVs are at their slowest, increasing the number of AGVs has a negative impact, this is caused by the time in which it takes AGVs to navigate around one another, thus when increasing the speed, the number of AGVs has a more positive impact as they are able to traverse each other faster.

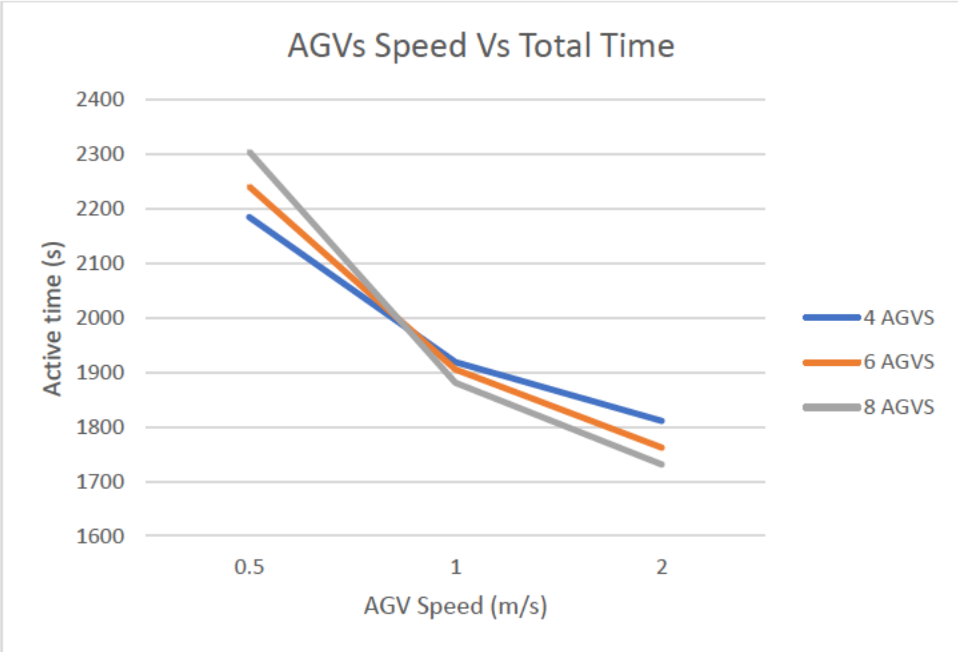

Figure 37 AGVs of varying speed and number vs the amount of time spent delivering parcels

Problems with the simulation:

Even though the system provided a much greater understanding as to how an AGV controller operates and the implementation of one, it was not without its own faults that could prove troublesome when attempting to implement an adaptative model.

- No Charge Logic

The way that the current AnyLogic system works is by making the AGVs return to base when there is no task for them. This is the current solution to the charging logic. This, however, does not apply in the real world as AGVs will run out of battery unless it is specifically managed and told when to go and charge. One example discovered is in a business called Orora, where the AGVs charge for two minutes every hour on high current chargers. This logic is built into the AGV controller.

- Collison and path logic

The collision and path logic seemed to be somewhat odd as sometimes an AGV would be resting in an Idle position until getting clipped by another AGV resulting in the other AGV going into an infinite spin until requested by the AGV controller to stop. It should be noted that this did not have any functional effect on the system.

- Path weightings

Another feature that is difficult to incorporate in the software is path weightings. Path weightings allow the AGVs to take paths that are normally blocked off to them unless their current path is blocked and less efficient to travel. Having path logic in a simulation allows much greater control when creating AGV logic as it allows for structured flexibility in the system. This is not achievable at all in AnyLogic or without advanced levels of knowledge of the software program.

- 3D models

Another problem encountered in the AnyLogic system is the ability to integrate custom models into the simulation. While it is not impossible to add custom 3D models to AnyLogic, it is not ideal as there are a few conversions that normal 3D models will need to go through. This is due to the fact that AnyLogic prefers to get its users to use their 3D objects.

- The number of items delivered.

Another test was done to calculate the throughput of the system based on the 1-hour time frame given by AnyLogic. This is shown in Figure 38 and Figure 39

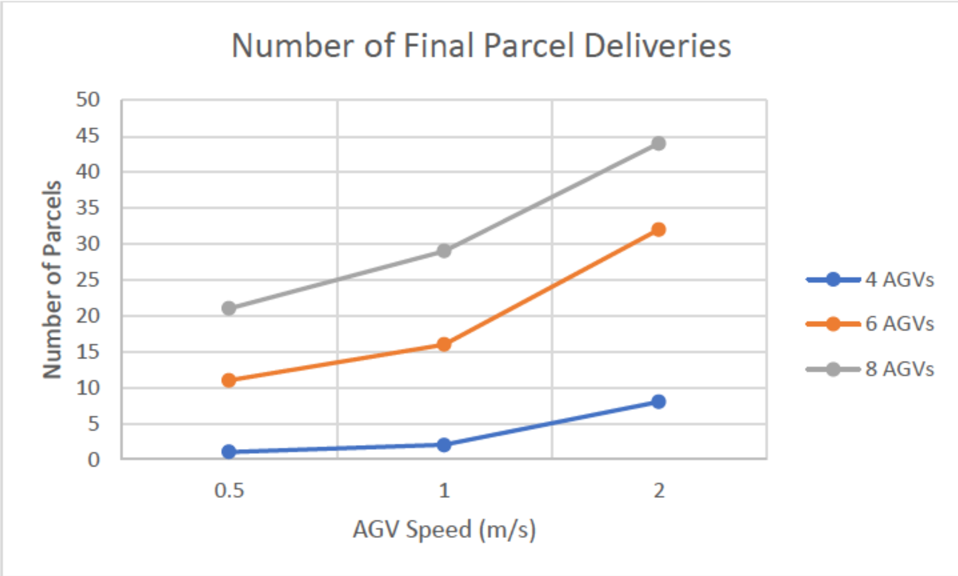

Figure 38 Total number of parcel deliveries within 1 hour of simulation with varying number of AGVs

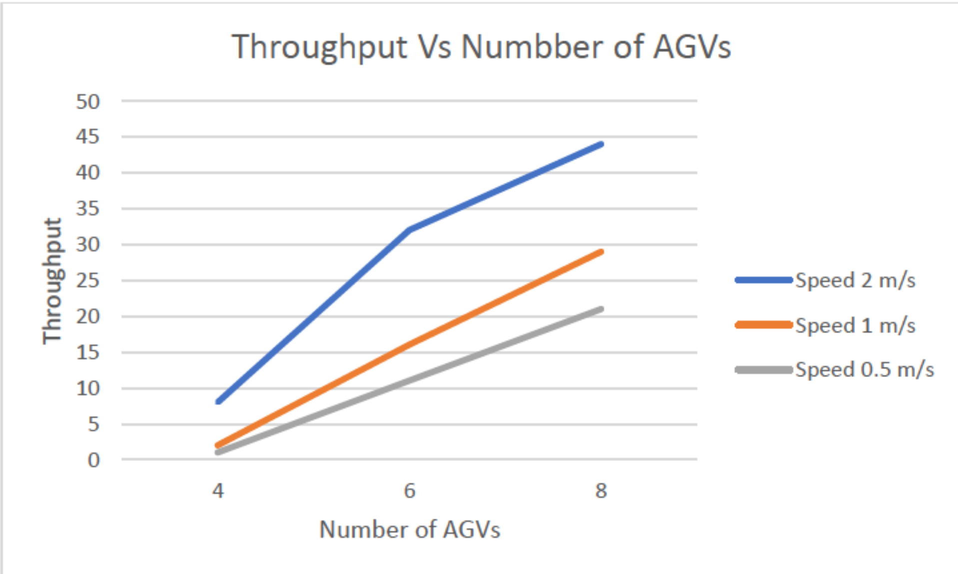

Figure 39 How speed effects throughput of the system with varying number of AGVs

It can be seen that as the speed and the number of AGVs is increased so as the overall throughput of the system did. The results however for the most part show a linear/ almost exponential growth particularly in Figure 38. Figure 39 however does start to show some plateauing when the simulation was run with the AGVs traveling at 2 m/s. The main limitation in this test is the fact that the simulation can only run for an hour thus making it difficult to find the optimal number when it comes to testing. This however was not the purpose of the simulation, which was to demonstrate how does an industry software handle an AGV fleet from which it did achieve this.

7.7 OPTION C:

Considering the knowledge gained from the previous phases between the previous Unity model, MATLAB and AnyLogic, a new AGV system is created. The goal of the new system is to compensate for the flaws found in previous systems.

- Control logic
    - State machine

The main method of controlling the system was determined to be a state machine. The reasons a state machine is the best choice for this kind of system is because of the way in which Unity calls the function frame by frame. As seen previously in phase one, when a state machine was not used, there were ample ways for Unity to skip frames and AGV commands, especially as the number of AGVs increased.

The state machine allows for the system to recover even if a frame is skipped as each AGV keeps track of its state, and if it is in the incorrect state, the system will normally fix this within a frame or two, meaning there are no noticeable control flaws on the system. Figure 40 below shows what this state machine looks like and the main logic flow of the system. This state is run every frame in order to keep the system up to date.

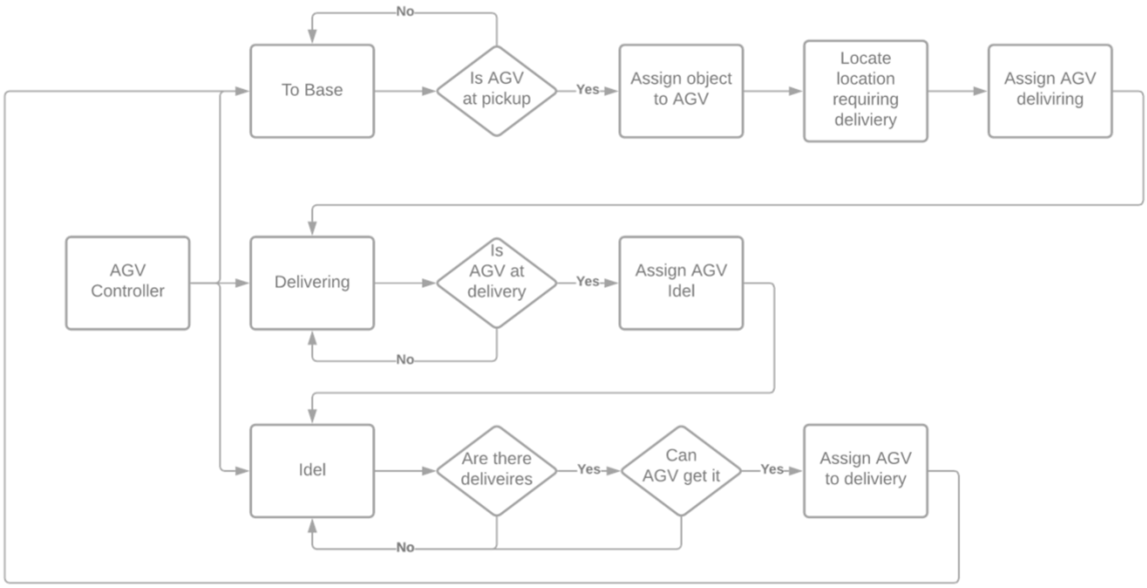

Figure 40 State machine logic for Option C AGV controller

When an AGV is moving to a certain destination, it is given a state as the AGV reaches said destination. The AGV then notifies the controller. Depending on the previous state, the system will designate the AGV as a new state along with a new goal location. This then allows for some of the workload to be taken off the AGV controller as the AGVs are directly put in charge of monitoring and notifying the system of state changes.

The system will iterate through every AGV and check to see if the AGV has changed state once per frame in order to prevent logic flaws that could end in deadlock, as seen in Option B. This method is better than the method in Option B as it does not rely on arrays that contain an AGV when it's ready to use and a separate array that contains AGVs in use. The system just looks at each AGV as it is, then assigning it a new task when it's complete.

- AGV Speed control – how it works

The new method of controlling the AGV also allows for a system to be integrated which allows the speed of the AGV to be controlled up to 100x, allowing for long term simulations to be carried out.

The time is integrated into the system by a method of using a universal clock from which all aspects of the simulation reference. What this allows for is the user to speed up the clock, which then directly speeds up all other elements of the system to match the new clock speed.

- AGV data tracking – how it works

Since the new system in Option C is far more streamlined than the previous Option B version, thanks to the help of the MATLAB and AnyLogic AGV system. Data tracking becomes simpler, as the AGV controller loops through each of the AGVs, the amount of time spent on that particular AGV state is captured. This allows for an overall system view of how the AGVs are performing and allows for the user to see if the amount of AGVs is too little or too much.

- Battery Logic

In a study done by (Hamdy n.d.) it was found that the AGV has a battery consumption rate of 0.0011 per ft which is then equal to 0.3 * 0.0011 = 0.00033 per meter. The constant is found to have 90% accuracy which is acceptable since the system is not mimicking specific AGV. It should be noted that this value compensates for a loaded and unloaded AGV thus creating the 10% discrepancy in the accuracy.

7.7.1 OPTION C RESULTS:

In order to prove the system functionality, an initial first test was created, where the object of the system was to deliver 200 parcels to 2 warehouses from a signal distribution centre. The distribution of the parcels followed a similar logic to that of the AnyLogic in Option D, where 30% of the parcels were allocated to one warehouse and 70% to the other. Figure 41 below shows the setup of the simulation.

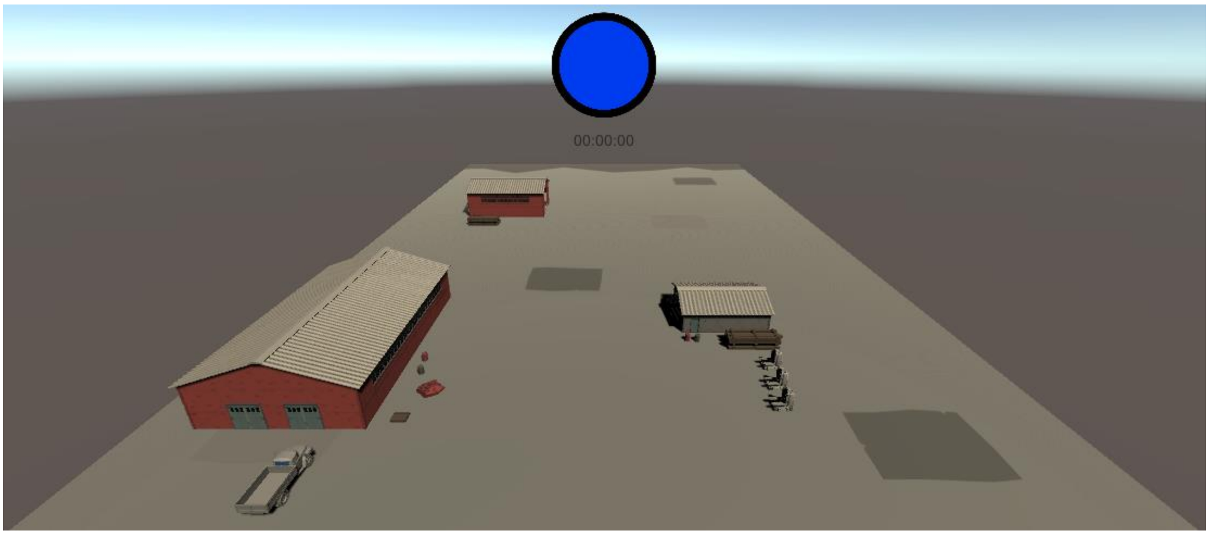

Figure 41 Option C simulation warehouses and distribution centre

Looking at Figure 41 above, it can be seen that the simulation is set up such that the middle has an increased waiting to it. What this means is that the AGV will not move into this zone unless other paths are blocked, and that becomes the quickest option based on weightings. This type of feature was not present in any of the previous simulations, thus giving this one a more flexible environment to work with.

The simulation was run with varying numbers of AGVs. The results can be seen below in Figure 42.

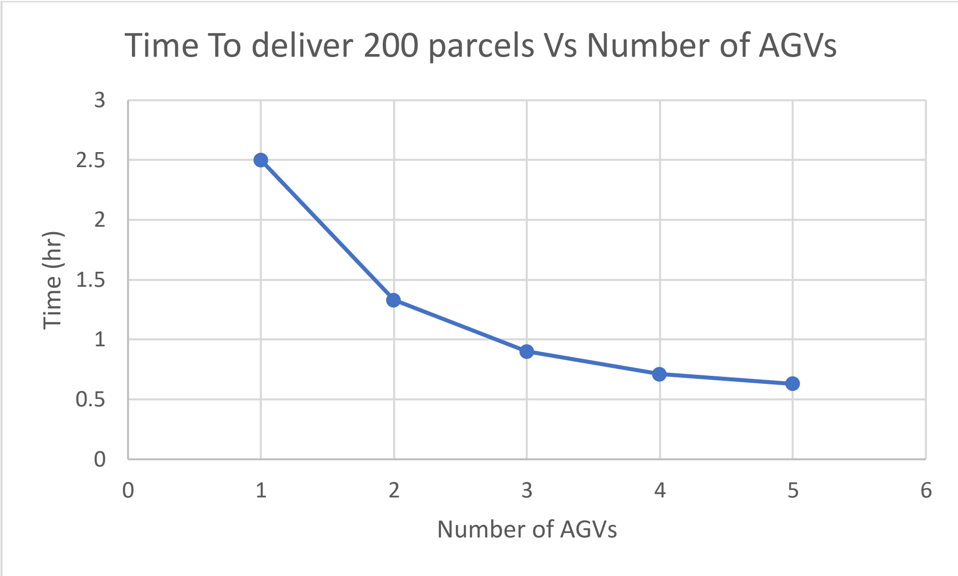

Figure 42 The amount of time the system took to deliver 200 parcels

As can be seen in the graph above, as the number of AGVs increases, the overall speed and performance of the system increased, allowing for the parcel to be delivered in a quicker time frame. However, the system does show that as the number increased, the graph starts to plateau. This is because the simulation is reaching its maximum number of AGVs, and the impact of adding more AGVs is minuscule.

This type of result is as expected as no AGV system is ever linear, even proven by the MATLAB Option D results where the system still had time increases with the AGVs being able to pass through one another due to the AGVs overcrowding and preventing other AGVs from delivering.

In order to mitigate bias in the simulation, it was run another 30 times with varying AGV numbers. However, this time the value between which drop off points being targeted were randomised, allowing for a more diverse delivery pattern. The results for this are shown below in Figure 43.

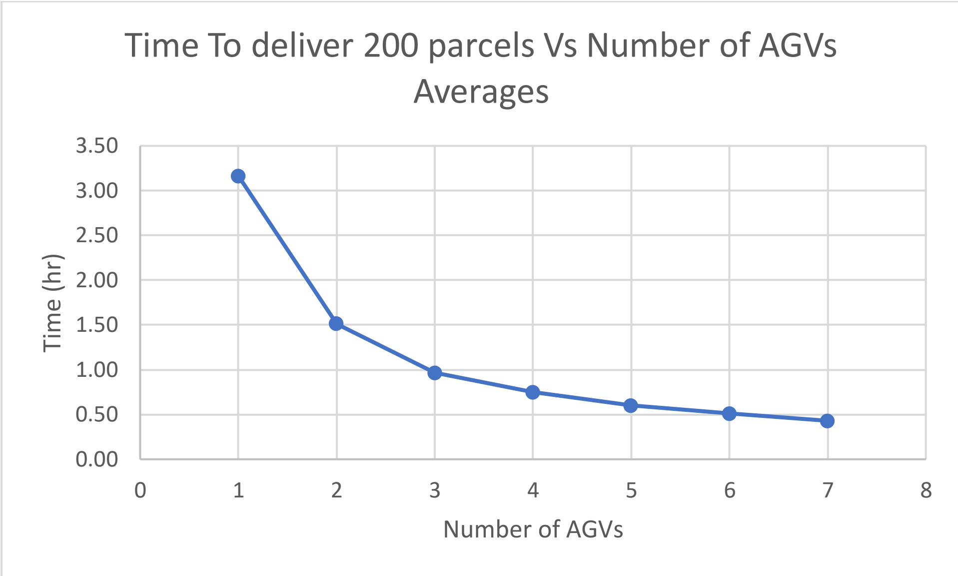

Figure 43 The amount of time the system took to deliver 200 parcels averages

Looking at the system above, it can be seen that the trend of time plateauing still holds true from the initial test in Figure 42 previously shown, thus showing that the system does start to behave as expected even with randomised outputs.

Another test of the system is to test the throughput. This is like the one conducted in AnyLogic, where the systems test how many deliveries can be done in 1 hour of simulation time. The test, as before, randomised the drop off points. The simulation is first tested when the AGVs have not dropped off time, thus resulting in the maximum number of AGVs the simulation can absolutely handle under the most optimal conditions. Figure 44 below shows the results of the simulation.

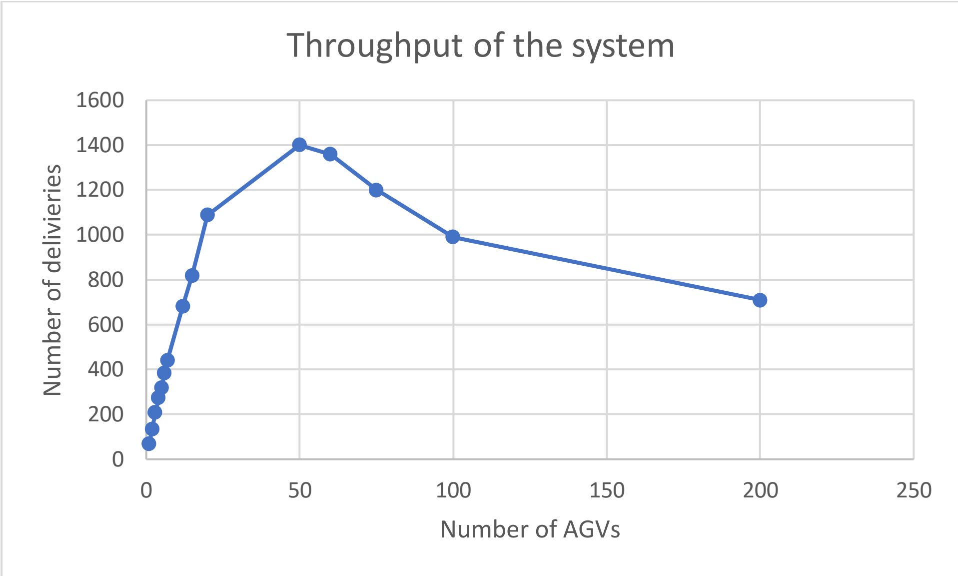

Figure 44 Throughput of the system between two warehouses with no wait time on the AGV

Looking at the simulation, it can be seen that the system behaves in a linear fashion until about 50 AGVs. After 50 AGVs, the system begins to drop off as the AGVs start to cause traffic jams resulting in AGVs unable to maneuverer around as easily. However, when setting the simulation to behave like a normal system where AGVs have to pickup and drop off products with an average time of 30 seconds for a pickup or drop off as found by (Azimi, Haleh & Alidoost 2010), the following Figure 45 is created.

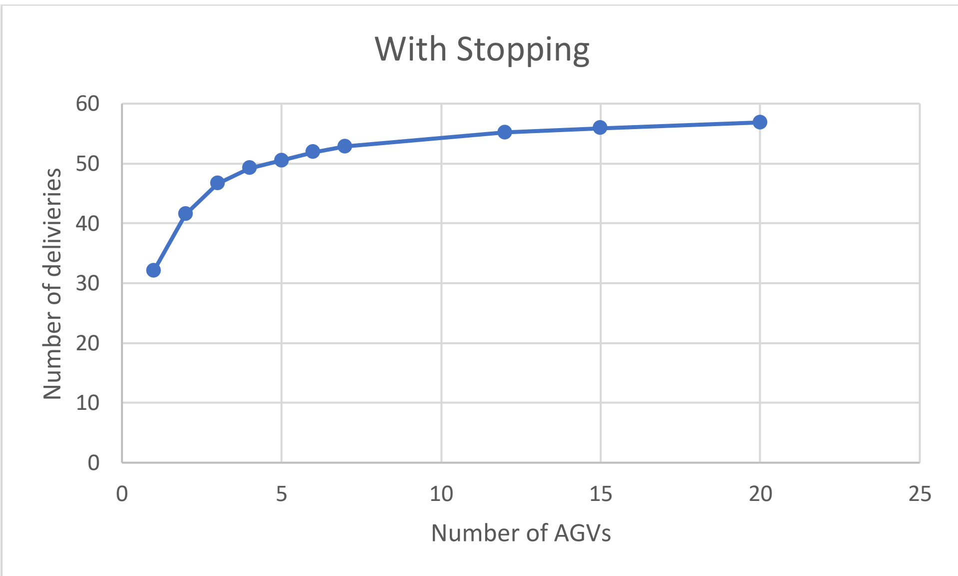

Figure 45 Throughput of the system between two warehouses with 30 second wait time on the AGV

As it can be seen, the system starts to behave as expected as the AGVs start to queue up the plateau occurs with far fewer AGVs which in this case is around 7 AGVs. This demonstrates that the system is starting to behave like a normal system, and the AGV controller is directing AGVs correctly and preventing deadlock, unlike the Option B and MATLAB simulations.

8 PRELIMINARY INVESTIGATION INTO HUMAN FACTORS

In order to gain an understanding of how AGVs can have an impact on people in a working with and amongst the AGVS. And attempt was made to incorporate human factors research into the project.

Incorporating human factors into the project required a questionnaire and ethics application was created. The ethics application focused on *"How can autonomous vehicles be effectively used for effective fleet delivery in manned and unmanned teaming of last-mile delivery vehicles and what is the impact on worker productivity, safety and trust in these emerging technologies?"*

The people to be interviewed were the defence company employees with experience in working in shipyard and recent exposure to relevant technologies in industry 4.0.

The interview was to cover key human factors related to technology acceptance including performance expectancy, effort expectancy, safety / trust in technology, culture, innovativeness and behavioural intention (Venkatesh et al., 2003).

Due to COVID-19 however there was great difficulty in conducting the interviews thus they were not included. In order to gain data, a pilot study was conducted with the defence company's Research and Technology team as well as an onsite visit to another small and medium sized enterprise (SME) ORORA Glass to see how they had integrated AGVs into their warehouse along with their current forklift drivers.

8.1 ORORA GLASS VISIT

One key aspect to the research into AGV integration with people was to visit a site that had accomplished a feature like this. One of the unique aspects of the ORORA Glass AGV integration was creating a separate warehouse where only AGVS where allowed. The AGVs stacked items form the production line.

What this allowed ORORA to achieves was take out the repetitive tasks of the drivers and let them focus on the more complex task of loading trucks and managing the weight distribution of the different glass bottle types.

When talking to the logistics team and the head forklift driver it was found that incorporating such technology had greatly reduces incidents of falling pallets and had reduced stress on drivers preloading pallets. Thus, this resulted in increased work health safety of the drivers and warehouse workers.

8.2 RESEARCH WITH THE DEFENCE COMPANY

Throughout the process, various meetings were held with key subject matter experts from the defence company. These meetings were to show the system as well as get feedback on the system.

The initial concept was to integrate into the current shipyard the AGVs however, as meetings progressed and the defence company was still planning the location of the facilities, It became a greater focus on having the system be more adaptable.

The system progressed from having the AGVs travelling around the shipyard to a more simple example.

When having the meetings with the R&T team, the key areas of discussion were mainly around the performance expectancy and effort expectancy.

8.2.1 PERFORMANCE EXPECTANCY

The performance expectancy of a system can be defined as the degree for which an individual believes that by using said system it will help the attain gains in their given job (Venkatesh et al. 2003, p. 447).

One of the performance expectancies of the system was that it was not too important as to what happens to the package once it reaches its destination, as stated by a participant below.

"I think the internals of the other buildings is not so much a concern because it is just not going to be that much space to be storing things in the other buildings, as soon as it gets dropped off, it will get used."

This shows that the system mainly focuses on delivering the components between the buildings as the defence company might try and do a lean manufacturing environment where the parts only arrive as needed. Another aspect of the simulation is that the level of graphic detail is not too much of a priority, but rather the logic of the system needs to be correct, as stated by a participant when asked about the different Visual elements that could be added to make the simulation look better.

"I am not a visual person. I do not care. The actual functionality I care about."

8.2.2 EFFORT EXPECTANCY

The effort expectancy of the system is defined as the degree of ease in using the system (Venkatesh et al. 2003, p. 450).

Effort expectancy is defined as the degree of ease associated with the use of the system.

in terms of the effort expectancy, it was made clear that it would be ideal to have a slider system that allows the various variables to be adjusted, such as the number of AGVs and speed.

"The important thing is to have all these things that we are talking about as sliders."

This will provide a more on the fly adaptability to the system allowing the user to make changes to the system in real-time. This is a consideration for future improvements to the project.

8.2.3 CULTURE

The culture aspect is the degree to which the company uses technology in order to make decisions. Culture of using the system can also be impacted by a social aspect by which degree the individual perceives that others believe they should use the system (Venkatesh et al. 2003, p. 451).

From the various general discussion with the R&T team, there is a strong culture towards simulation as a method of making decisions; however, the level of the impact varies between the different aspects of the shipyard. When implementing an AGV system, the simulation plays a very large role as it allows for a value to be obtained as to how many AGVs will be needed, the speeds for which the AGVs can travel and various other metrics that the team is interested in.

Simulation technology also seems to be picking up in the shipping industry as it allows for cost-effective rapid prototyping without the need to manufacture components right away.

8.2.4 INNOVATIVENESS

Innovativeness is an idea or product that features new methods; advanced and original (‘innovativeness’ n.d.).

From discussion with the R&T team, there is potential for innovativeness in modelling in the shipyard; this is further backed up by creating various research projects that focus on simulation, such as the one shown in this thesis. The R&T team has various other projects that focus on simulation as a method of creating more understanding for employees and creating visualisations for onsite visitors and government officials overseeing the project.

8.2.5 BEHAVIOURAL INTENTION

Behavioural intention is the extent for which user will use the given system in their job (Venkatesh et al. 2003, p. 456).

Talking with the R&T team during the project discussion about the system showed that the simulation developed could be used by the defence company; however, it would not be used to make a final decision as this would be the AGV manufactures' job their industry level AGV software.

8.3 FUTURE PROGRESS

Due to the States and business opening up now future progress can be made by going and performing the original interviews on the defence company employees with experience in working in the shipyard. These interviews are the most crucial as it foster an increased understanding of how these system can be integrated as well as the impact it will have on those who. Will actually directly interact with a system like this.

9 DISCUSSION

9.1 OPTION B:

Phase one of the project was the initial attempt at understanding the problem and using unity to create a solution. Option B allowed for a greater understating of AGV systems and how unity works and how data and objects are created and based between various scripts. The key takeaway from Option B is how crucial scheduling and AGV increases are on a system and that if the logic in a system is flawed, it can result in extremely large delays and large traffic jams.

9.1.1 AGV SCHEDULING DURING IDLE TIMES

The paper A framework for selecting idle vehicle home locations in an automated guided vehicle system (C-H & Egbelu 2000). Dwell points were created for AGVs. These points are intersections that have the optimum distance between all pickup points in their area. Using this methodology, AGVs have the shortest distance possible on average when assigned a task. An illustration of this can be seen in the Figure 46 below.

Removed due to copyright restriction

Figure 46 Location to store AGV during idle times for (C-H & Egbelu 2000) system

To develop a system like this, key pickup locations will need to be able to, in theory, must be able to hold the given number of AGVs in the case of no deliveries required. However, one of the key problems with this simulation is that they do not specify when the AGV should move back to the Idle base but instead take a mathematical approach using heuristics to calculate the optimal idle position in a system with the optimal solution being found 84% of the time. (Moussa 2020) on the other hand in the article Manufacturing 4.0 Operations Scheduling with AGV Battery Management Constraints describes how to tell when the AGV should go to an Idle location based on the current battery charge of the system. This type of system is modelled on an AGV operation that is nonstop thus requiring careful management of the AGVs battery level when delivering and picking up objects. Thus, it is assumed the AGV is never idle. Using the conditions below, a cost function is used to calculate the optimal charging location while the AGV is in transit, based on the various conditions the AGV could be in.

1. The nearest battery station.
2. The farthest reachable battery station on the current route.
3. The first battery station encountered on the current route.
4. The battery station leads to minimum delay.

In a study by (Leite et al. 2015), they simulated a plant using an AGV system in an industrial environment and analyses the advantages and disadvantages of the project. The simulation can be seen below in the Figure 47

Removed due to copyright restriction

Figure 47 (Leite et al. 2015) simulated plant

In the simulation, the system used 7 AGVs from which the average Idle time for all the AGVs was 0.16%, with an approximately 5% downtime of the system and 95% overall utilisations. This can be seen in the Figure 48 below. When comparing this with the results obtained from Option B, it is clear that the Option B ship simulation has extremely different results where a value of over 30% idle time is achieved. Comparing this to the 0.16% found in an industry-standard simulation, it can be seen that the model is flawed. The Option B model also achieved an overall utilisation of 70%, which is 25% less than the industry standard model shown above.

Removed due to copyright restriction

Figure 48 Percentages of AGV use in (Leite et al. 2015) study

The Option B model can be seen to be missing some key features as outlined in all three of the simulations. It can be seen that by potentially applying the methodology from selecting idle locations and a battery management system, Option B could potentially reach a utilisation rate of 95%.

9.1.2 WHAT SHOULD BE THE EFFECTS OF INCREASING AGVS IN A SYSTEM

When looking at the paper written by (Lyu et al. 2019b) where the paper has a main focuses on AGV scheduling in a flexible manufacturing system by simultaneously considering the optimal number of AGVs this means the study will look at the effects of a system in parcel delivery time with respect to the number of AGVs added into the system.

Looking at the graph below in Figure 49 as the number of AGVs in the system increased, the system's performance increased. Looking at the Figure 49 when there is only one vehicle in the system, the waiting time for the AGV is zero as the system 100% utilises the AGV; however, the overall time that it took to deliver all parcels is at its peak. When there are 7 AGVs in the system, the operations are performing at their peak. However, the waiting time for the AGVs is also at its peaking, meaning that the AGV utilisation is extremely low, thus adding lots of unnecessary cost to the system.

Removed due to copyright restriction

Figure 49 (Lyu et al. 2019b) AGV number vs time for deliveries to complete

The key difference when looking at this data as opposed to the data from Option B in the Figure 18 above is that as the number of AGVs was increased in Option B, the system did not react in the same manner. The system behaved identically as more AGVs entered the system except for the added congestion at certain delivery points. One of the core reasons for this is due to the system locking up, causing AGVs to not finish performing deliveries due to the congestion. The systems Idle time also does not have the same response as there is no exponential growth occurring but rather a continuous 30% idle time no matter the number of AGVs utilised.

It can be expected that the data of the Option B system should follow the trend identified in the Figure 49 above when it comes to the time to complete tasks based on the number of AGVs.

9.1.3 REALTIME SYSTEMS VS SPED UP SYSTEMS

When comparing Option B of the project to other simulations, the time for which the simulation runs come nowhere close. Looking at (Leite et al. 2015) where the AGVs were utilised for 90, in order to accomplish this a simulation would require a system that can adapt in speed; otherwise, in order to get any valuable data out of the system, a person would have to wait 90 days for the simulation to complete. At its current state phase, one simulation is only slightly faster than a normal AGV, resulting in a simulation time of approximately 75 days to simulate the same system. This is extremely undesirable for a system and is a key aspect of the simulation that needs to be addressed.

One of the key differences in the simulation models is the software used for the simulation. The previous examples in the discussion all use industry software allowing for large time scales to be simulated. One way to replicate this kind of process the system clock for unity will need to be scaled while then converting all aspect of the simulation to the function of the new clock, thus allowing the speed of AGVs and battery calculations to work faster.

9.1.4 OVERALL DEDUCTIONS

The overall deductions from Option B are that the system needs major improvements to prevent deadlock and allow for the user to control the speed of the simulation allowing for greater control. In order to do this, these real systems are to be investigated to understand how the industry approaches these problems.

9.2 OPTION D:

Option D focuses on adapting and learning how pre-built systems operate and how to incorporate those features into unity. This phase is split into two aspects: MATLAB and the other on an industry software called AnyLogic.

MATLAB

In MATLAB, a simulation of LineZero at Tonsley is used to move parcels from 1 pickup point to three drop-off points. Through the use of occupancy grids AGVs are programmed where they can travel.

9.2.1 COMPARE TO LITERATURE REVIEW SOURCES WITH AGV INCREASE ON THE SYSTEM.

When comparing the MATLAB AGV system to the one created by (Lyu et al. 2019b) in the paper "approach to Integrated Scheduling Problems Considering Optimal Number of Automated Guided Vehicles and Conflict-Free Routing in Flexible Manufacturing Systems" seen in Figure 28 above, it can be seen that the system does not follow the same curve but rather decreases until it reaches 4 AGVs then the system starts to increase as more AGVs are added. As opposed to the data in Figure 49 which starts to plateau as the AGV number increases showing no further performance improvements after approximately 4 AGVs are added. This plateau is what is to be expected out of the MATLAB AGV system; however, this is not the case. There are two main causes for this issue, as discussed in the results. The first reason is the choke points in the system, which in this case, the main choke point is the pickup area where all the AGVs gather waiting to be assigned a parcel. The second cause is the method for which the AGVs use to traverse each other, that being that one AGV will be brought to a stop and the other AGV being allowed to move through it. What can happen is that the AGVs can take a while to traverse through a large number of AGVs as the system will give one priority to move then the next and so on until all the AGVs have moved. Figure 50 below explains this situation with a simple example.

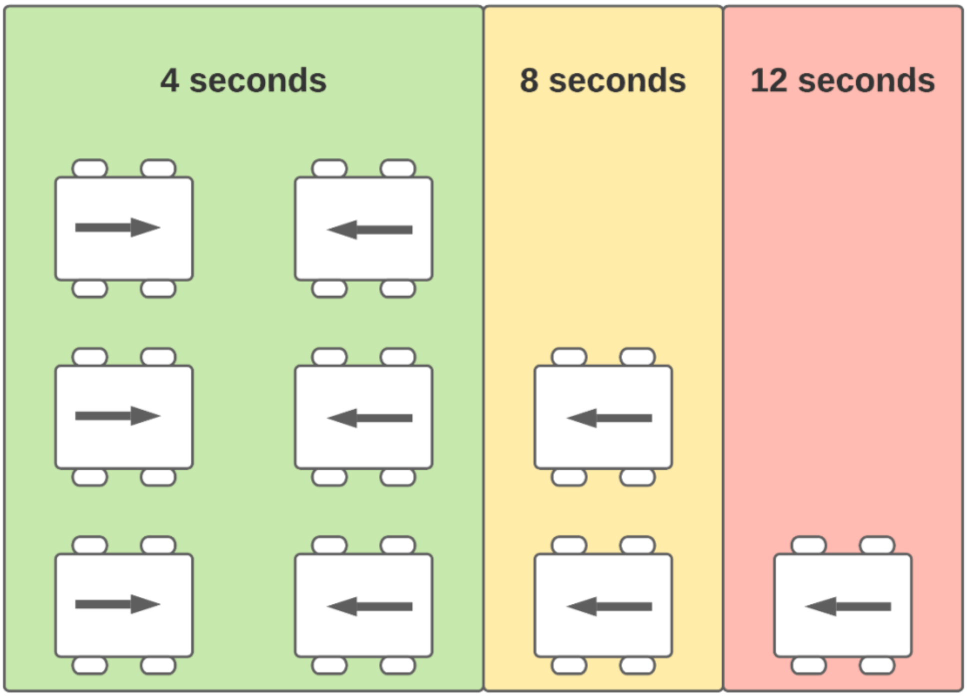

Figure 50 MATLAB time increase as number of AGV increases

Looking at the Figure 50 above, it can be seen that an AGV takes 4 seconds to pass another AGV due to stopping and organising which AGV gets priority each time. This means that as there are more AGVs introduced into the system navigating each AGV takes longer. Thus, it can be seen that in an AGV system, it is crucial for AGVs to have the ability to navigate around each other, thus reducing this time delay and allowing the system to behave like the one in the Figure 49 above. This type of system will prove troublesome if a business is to expand or get more orders than usual as it will just cause all the orders to take longer when trying to use more AGVs.

9.2.2 AGV SCHEDULING DURING IDLE TIMES

The AGV system works by taking all available AGVs and programming them move to the distribution location. This results in the distribution location becoming the waiting area for AGVs to get assigned packages. Comparing this to other systems such as the one made by (C-H & Egbelu 2000), where the AGVs are assigned waiting areas that give them the optimal distance to the various loading zones, the performance is far below expected. As can be seen, by the data obtained from the Option D MATLAB simulation, it is not ideal to have a system that does not have a waiting zone for the AGVs as this causes the system to perform worse as the number of AGVs is increased due to traffic jams as shown.

9.2.3 CHOKEPOINTS AND DEADLOCKS IN AGV SYSTEMS

Due to the path size is given for the AGVs in the simulation, it can be treated the same as a bi-directional AGV simulation. When comparing the simulation to other bi-directional systems, it can be seen that there is no method for handling deadlocks and blocking. The paper "Shortest Routing of Bidirectional Automated Guided Vehicles Avoiding Deadlock and Blocking" by (Wu & Zhou 2007) shows that adding a method compensates for a deadlock can exponentially help a system. The paper shows that one of the best methods for preventing deadlock is to calculate all AGVs paths that are active before assigning the current AGV a path. By using this method, the AGVs are able to manoeuvre far more effectively as priorities are given even before the AGV moves, allowing for a system in Figure 50 to behave in a manner where the AGV going the opposite direction to the three AGVs could just pass through all three at the same time without recalculation.

Another major aspect between the MATLAB Option D simulation and other AGV models is its method of handling AGV collisions such that the system allows AGVs to pass through one another. Other models do not have this feature as it causes simulation issues and reduces the overall accuracy of the system significantly especially if the simulation runs for a three-month time frame.

The paper "Multi-AGVs Collision-Avoidance and Deadlock-Control for Item-To-Human Automated Warehouse", written by (Yan, Zhang & Qi 2017b), discusses the impact of deadlock in an AGV system. The paper proposes various ways to try and mitigate deadlocks by proposing rules and various control strategies. Below in Figure 51 is the map of the warehouse and the graphs obtained in their study on deadlock.

Removed due to copyright restriction

Figure 51 Grid warehouse layout in (Yan, Zhang & Qi 2017b) study

Removed due to copyright restriction

Figure 52 impact of AGV amount in relation to through put for (Yan, Zhang & Qi 2017b) study

Looking at the graph in the Figure 52 above, it can be seen that as the number of AGVs enters a system, the average delay increases. What this shows that the MATLAB Option D model is somewhat correct with its values as the system responds. In the same way, the more AGVs, the greater the delays. However, the key difference is the system proposed by (Yan, Zhang & Qi 2017b) has controls in place to mitigate the effect and overall prevent the deadlock. This is proven by the system handling 40+ AGVs while the MATLAB system cannot run a simulation with greater than 6 AGVs. One of the key aspects of the simulations that allows it to avoid deadlock is to control the AGV and control which lane is accessible to the AGVs. What this means is that the AGV cannot take a path that has already been claimed. This method does result in some AGVs travelling a greater distance; however, overall it prevents collisions and deadlock while also reducing the total time.

The AnyLogic Simulation focused on a small warehouse simulation in order to understand the impacts of AGV fleet size increase on throughput as the simulation could only run for 1 hour, thus making throughput a key result from said simulation.

9.2.4 COMPARE TO LITERATURE REVIEW SOURCES WITH AGV INCREASE ON THE SYSTEM

Comparing a similar study done by (Valmiki et al. 2018), where the throughput of a system was compared to the number of AGVs. The following results in Figure 53 were achieved from their simulations in a small manufacturing environment.

Removed due to copyright restriction

Figure 53 (Valmiki et al. 2018) warehouse study and AGV throughput

Looking at the Figure 53 above, it can be seen that as the number of AGVs increased, the system had a higher throughput until approximately 15 AGVs. After this point, the system started to plateau, which has been seen in previous models.

When comparing this to the AnyLogic throughput, there are some similarities when taking the scale difference of the simulations into account. As this simulation is on a smaller scale inside a factory-direct throughput comparison is not possible. However, some trends can start to be seen in the AnyLogic data.

The main trend that can start to be noted on the Option D AnyLogic simulation is when the speed is set to 2 m/s as the AGVs start to come closer to the limits of the system, a small curve is almost observable. This is as expected as AnyLogic is an industry software and is designed to manage AGV fleets at an industry level.

The simulation, however, is only for learning and understanding purposes; thus, the level of depth that the simulation can achieve is nowhere near the scale shown in the Figure 53 above. The key differences between the simulations are the number of AGVs that the simulation is designed to handle, the AnyLogic has the ability to simulate 8 AGVs with the current configuration while the other simulates is shown to simulate at least 20 AGVs.

9.2.5 AGV SCHEDULING DURING IDLE TIMES

When scheduling the AGVs, the system makes the Idle AGVs return to the closest available home base, similar to the system shown by (Moussa 2020). The key difference, however, is that the Option D AnyLogic does not take the battery or other external factors such as battery life of the vehicle. The furthest reachable battery station on the current route. The first battery station encountered on the current route or the battery station that leads to minimum delay into consideration when choosing the idle location. This is mainly because the simulation does not factor in charge of the AGV, which is a major flaw addressed in (Moussa 2020) simulations. Since the AnyLogic simulation only runs for 1 hour due to the educational use limits, it is not possible to simulate these effects of battery on the system. (Hamdy n.d.) discusses that this is a common problem in many AGV simulations where the battery of a system is not taken into consideration

9.2.6 CHOKEPOINTS IN AGV SYSTEMS

As previously mentioned, chokepoints have been a key problem with the AGV simulations. The Option D AnyLogic system handles these chokepoints very well. One of the key reasons is the ability for the AGVs to easily navigate each other based on the logic that is used in the AnyLogic free roam AGV feature. Since the simulation does not require the AGVs to follow a particular flow path, it allows the system to calculate many different paths around a particular object. This being said, there are still some choke points in the shelving area of the system as it provides a small area for many AGVs to traverse. The open plan AGVs navigate the system just as efficiently as flow path AGVs in terms of overall time as when the AGVs are in closer proximity. More care is taken when manoeuvring one another, while the open space navigation makes up for any time delays caused by the congestion in the system.

9.2.7 DETERMINISTIC SYSTEMS

The results of the AGV system are deterministic, meaning that there is no randomness. This was tested by changing the various values in the system, and no change was noticed when using exactly the same values between runs. One of the key reasons that a deterministic system can be beneficial is that it can prevent a system's deadlock. This is used in database systems often (Ren, Thomson & Abadi 2014); however, it can apply to AGVs system as manoeuvring through a system can be similar to navigating databases in particular graph databases. However, determinist systems also come at disadvantages. More pre-calculations are needed to run the system; thus, initial loading and start-up times can take a while.

9.2.8 OVERALL DEDUCTIONS

The overall deductions that can be made is that the collision logic on the AGV needs to account for more than just one move into the future as the system will end up in a deadlock, as demonstrated by the MATLAB code. The system should also be robust enough to prevent AGVs from travelling through walls and clipping each other. It can also be seen that battery charge is a key feature that is missing from the two AGV systems and is a necessary feature for simulations to run for more than 1 hour. The AnyLogic system behaved as expected but lacked data due to simulation time frame and the initial setup of the system with a number of AGVs but overall served its purpose in providing a deeper understanding of how fleet management is done in an industry setting and the types of commands and states that an AGV can be in.

9.3 OPTION C: UPDATED AGV CONTROL

9.3.1 AGV SCHEDULING DURING IDLE TIMES

In the Option C simulation, the AGVs Idle time is handled differently from that in the AnyLogic and in the system (Moussa 2020). In the Option C simulation, the AGVs currently return to their original Homebase, where they are spawned. When the AGVs are at the base, they also start to charge. However, one of the key issues with this design is the fact that the AGVs will always spawn in one location. Thus extra care needs to be taken into account when selecting the base location as it needs to be in the optimal area as shown by (C-H & Egbelu 2000) in Figure 46 above. Some problems that can also occur from the Option C model is the fact that the user has no choice to try and diversify charging locations that will allow for unique charging logic to take place, such as the list below found in (Moussa 2020) paper.

1. The nearest battery station.
2. The farthest reachable battery station on the current route.
3. The first battery station encountered on the current route.
4. The battery station leads to minimum delay.

In order to incorporate something like this into the system, a process can be put in place where the user can drag locations on the map, and the software will automatically populate that area with the optimal number of AGVs that can comfortably fit in said area.

9.3.2 BATTERY CHARGING

As stated before, each AGV is spawned with its own charging station. This is not an ideal situation as it increases purchase cost, maintenance cost, and space that the system takes up. When looking at companies such as ORORA, the head of logistics stated that the ratio is 0.42:1 for charging stations (Watson 2021). Figure 54 below depicts this relationship.

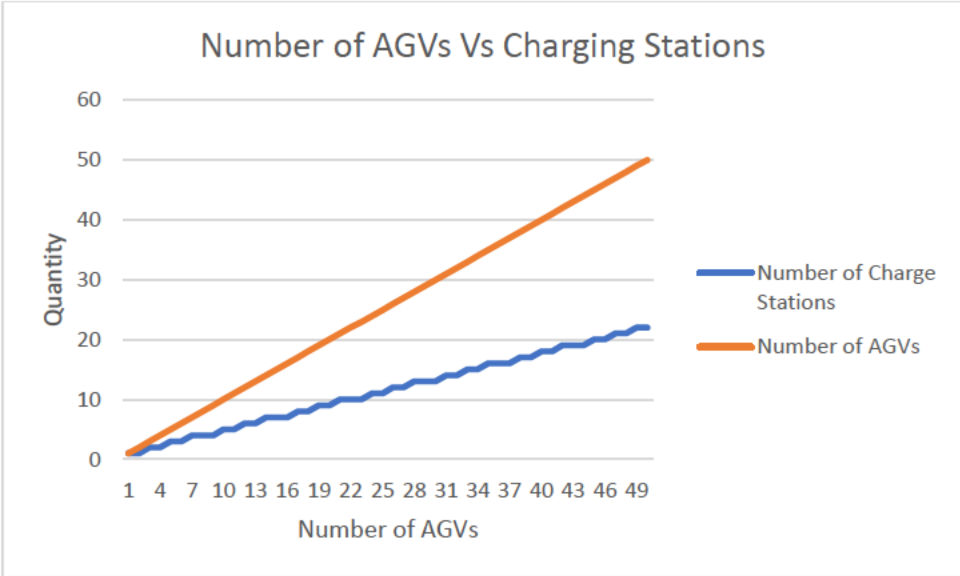

Figure 54 Industry charging station to AGV ratio

This type of situation is ideal, especially in 24/7 manufacturing environments, as it is extremely unlikely that all AGVs will need to be charged at the same time, and even if that is the case, the AGVs should not go to charge when they are just about to run out of battery as allowing the battery to drop below 20% is not good for the system. Thus, the AGV should always have at least 20% battery, thus giving it time to wait or finish a delivery while waiting.

In order to create a simulation that follows this relationship, the charging logic will need to be adjusted. According to a paper by NEC Energy Solutions ('When-it-comes-to-powering-your-AGVs-whitepaper-FINAL.pdf' n.d.), there are two main types of charging methods. The first method is to charge the AGV for 2 hours every 6 hours, thus resulting in 75% uptime for the AGV. The second method is to charge for ten minutes every 5 hours, resulting in 97% uptime. It is specified that having a battery that can charge fast, i.e. high current capable battery, will be better for the system in the long run. The latter method is also used by Orora for charging their AGVs with slightly stricter controls where the AGV has to charge for two minutes every hour.

This type of logic can be added to the system by randomising the charging time between all the AGVs every hour. The AGVs can then be stored in an array that the AGV controller checks. If the AGV is doing a task, it can allow it to finish then send it to charge once complete allowing for full completion of tasks and allowing the AGV not to have to charge at exactly the dedicated time. Another method would just be to divide the hour by however many AGVs there are; thus, if there are 6 AGVs every 10 minutes, a new AGV can be sent to charge for two minutes.

9.3.3 CHOKEPOINTS AND DEADLOCKS IN AGV SYSTEMS

This new system is vastly better than both the Option B and Option D MATLAB simulations when it comes to handling deadlocks and choke points. This is due to the system's ability to avoid obstacles as well as because it functions as a free roam AGV, allowing it to traverse as it sees most optimal. The main choke point of the system occurs at the delivery points as the number of AGVs increase, as demonstrated in the throughput Figure 43 above. The system also performs better than some of the other AGV examples listed above because of this fact.

9.3.4 NUMBER OF AGV INCREASE ON THE SYSTEM

When comparing the behaviour of the system to other studies, it is clear that similar trends start to exist. When looking at Figure 53 above on the system's throughput of a similar size from the study by (Valmiki et al. 2018), it is clear the trend and flatting of the curve are very similar. The study did manage to achieve higher throughput for the AGVs, but this is because of various factors. One of the core ones would be tuning the AGV pickup/drop off times and the speed at which the AGV can move, which are not clarified in the study. The study also considers and focuses on adjusting variables such as the carrying capacity of each AGV, buffer size at machining stations, number of pallets, and number of AGV's.

In order to make the system behave more like the one shown in the Figure 53 the carrying capacity should directly affect the speed of the machine as well as allowing the user to set up buffers on the various drop off points.

The other comparison of the system is with the study conducted by (Yan, Zhang & Qi 2017b) where the throughput is also calculated and the graph in Figure 43 above has a similar output to Figure 53 What this shows is that the system is starting to simulate the basic output of other systems from various studies showing that the state machine switch was a success over the other more linear method of organising the number of AGVs. Another factor that should be noted is that by using the state machine, the number of AGVs can be dramatically increased given the correct area size in the simulation.

However, the key difference when comparing the AGVs systems is that the current Option C model is not as organised as the ones shown. This is partly due to the system being designed from scratch and, in particular, to be flexible, while the other systems are for known situations with pre-existing software. The scales of the operations are also different. A greater focus was placed on the trends of the data and not the raw values, as AGV behaviour, when it comes to fleet size increase, is fairly universal.

9.3.5 OVERALL DEDUCTIONS

Overall, Option C behaves like other models in previous studies where the AGV fleet size trends match the previous studies. The model is adaptable as the various pickup locations and drop off locations that can be changed along with the number of each of the aforementioned locations. The number of AGVs can also be changed while allowing for the simulation time to be either increased or decreased as well as the ability to control the number of parcels delivered or the amount of time the simulation can run for with simple code changes.

However, there are still limitations to the simulation as it does not consider a specific type of AGV nor does it assume the AGVs will break down. The current system also works off spawning a charging station per an AGV in order to provide a return point, this should be optimised to match the research given by (Moussa 2020) as previously mentioned. There is also no consideration of the operations of working with manned versus unmanned machines as the simulation logic would be the same for a manned forklift vs unmanned one.

This simulation however does show that incorporating AGVs into a manufacturing environment can increase productivity as the system is extremely predictable and can be simulated to get accurate results.

10 CONCLUSIONS AND FUTURE WORK

Overall, it can clearly be seen that incorporating AGVs into a company proves beneficial as shown through the literature review and the various software that is designed to show business how implementing AGVs and automation can improve productivity.

The system was able to function as expected by doing the three-phase process where the final AGV system was able to simulate an AGV system where the number of AGVs could be set as well as having the ability to move/add drop off and pickup locations.

There are however some limitations to the system where the AGVs do not break down and the system does not consider interaction between manned forklifts and the AGVs, and the system cannot be used for final quotes on AGV systems due to these limitations. There is still future work that needs to be done in order to improve the system. Below are suggestions that should be conducted and considered for further research:

1. A key aspect that was incorporated in the AnyLogic system was a heatmap. Incorporating the heat map will add valuable information to the system. By incorporating a heatmap the user will be able to see key points in which AGVs are spending most of their time whether that be due to queuing or waiting to bypass other AGVs. The knowledge gained by a heatmap will allow for change to occur that will better the system overall
2. Another aspect that would add value to the system is to incorporate pedestrians. By adding pedestrians, the time values for AGVs will become more accurate while also allowing for the user to see the impacts of certain foot paths to the AGV systems.
3. Further research can also be conducted towards finding the impact of different material handling and AGV choice. Idle the system will allow for the user to select an AGV from a predefined list as well as an average parcel weight thus increasing the accuracy of the system especially when it comes to battery usage and charge times.
4. One of the key components that is missing is a simple user interface that will allow for the various locations and drop-off points to be selected without having to reconnect the new location in code. This can be done by allowing the user to select if they want

to place a drop off point or pickup then detecting where they click and apply that as the point in the system.

This project has been an exceptional learning experience in both programming as well as the knowledge gained in the process of developing an AGV fleet control system that needed to operate under various conditions while keeping track of all the AGVs. The project has also demonstrated the complexities of having to integrate said systems into a manufacturing environment.

11 APPENDIX

11.1 TYPES OF AGVS

('Automated Guided Vehicles (AGV) Meaning & Types' 2019)

Automated Guided Carts (AGC): These are the most basic of AGVs and do not normally have many features. The systems can use magnetic strips or a large variety of sensors allowing this AGV type to transport a large variety of materials. These AGVs are often used for sorting, storage, and cross-docking applications

Forklift AGVs: These AGVs are another very commonly used AGV which focus on preforming the tasks of moving pallets like normal forklifts.

Towing AGVs: To move nonpowered trailers similar to trains Towing AGVs are used. These AGVs normally move heavy loads across large distances. This type of AGV is often called an autonomous train.

Unit Load Handlers: These AGVs are designed to carry discrete loads such as individual objects and pallets.

Heavy Burden Carriers: The AGVS are designed to carry very large loads which are for large scale assembly. These are a niche type of AGV and can vary depending on the industry and large object being carried.

Autonomous Mobile Robots (AMRs): This is the most intelligent AGV as they are equipped with large amounts of sensors and intelligent navigation systems. These types of AGVs are normally used to navigate complex environments such as a shipyard. The unique aspect of these AGVs is that they do not require much change in infrastructure thus can be intergraded into a workplace more easily. Having a s system like this could be ideal for a place like Shipyard where there is still uncertainty where buildings and locations of things will be.

'

11.2 OTHER SIMULATION SOFTWARE

Some of the other software that can be used are Gazebo and Unity. These can be used to create accurate 3D simulations. Gazebo has a strong focus on robot control, while unity has a focus on creating dynamic environments with the intent of video game development.

It should also be considered that there are software's that are used by industry to simulate AGVs. One very notable piece of software is AnyLogic ('AnyLogic: Simulation Modeling Software Tools & Solutions for Business' n.d.). AnyLogic is a software that is used for optimising and designing warehouse layout and operations. The tool is used by many industries to provide powerful simulation models that can provide information on forecasting and current operations. The simulation software comes with many libraries and models such that it can be intergraded with many different industries such as rail logistics, oil & Gas but mostly in supply chain and logistics.

Another piece of industry software is FlexSim ('3D Simulation Modeling and Analysis Software' n.d.). This software is used to model 3D native environments, FlexSim also has factors in place that can provide more realistic variability in simulations using a large data set of statistical distributions and random numbers. The key difference between AnyLogic as spoken about above is that FlexSim does not offer the same number of features for the student version as AnyLogic does.

ARENA ('Arena Simulation' n.d.) is another simulation software the competes with FlexSim and AnyLogic. Arena specialises in modelling process flow in order to determine bottle necks and inventory problems in a factory. The software uses block logic in order to create the simulations.

11.3 INTERVIEW LETTER OF INTRODUCTION

Dear Sir/ Madame

I hold the position of Associate Professor in the College of Science and Engineering at Flinders University.

This letter is to introduce Mr Richard Ellis who is a Master of Robotic Engineering student in the College of Science and Engineering at Flinders University.

His research is part of a project entitled “Developing A Flexible and Adaptive Shipyard Fleet for Last Mile Delivery”. The objective of this project is to explore the effective use of autonomous technologies to transport goods in shipbuilding as part of the warehouse and distribution strategy, that can involve autonomous ground vehicles, drones and forklifts.

You are invited to participate in an interview that will require approximately 30 minutes of your time (up to an hour). The interview will be undertaken face-to-face at Tonsley.

Be assured that any information provided will be treated in the strictest confidence and none of the participants will be individually identifiable in the resulting thesis, report or other publications. You are, of course, entirely free to decline to answer particular questions.

The research team anticipates no risks from your involvement in this study. We endeavour to reduce any inconvenience to you by conducting the interview at a time and place of your choosing, wherever possible. However, if you have any concerns regarding anticipated or actual risks or discomforts, please raise them with the research team.

Participation is voluntary and you may refuse to answer any questions or withdraw from the study at any time without effect or consequences. A consent form is attached and must be returned signed prior to the interview taking place. If you have any questions about the project or your participation in it, please contact Russell.brinkworth@flinders.edu.au for more information.

Interview findings will be analysed and presented in the Master Thesis written by Richard Ellis. If you agree, an email with a link to the report will be linked to the participants.

Thank you for your attention and assistance.

Yours sincerely

Dr Russell Brinkworth

Associate Professor

College of Science and Engineering

This research project has been approved by the Flinders University Social and Behavioural Research Ethics Committee (Project number XXXX). For more information regarding ethical approval of the project the Executive Officer of the Committee can be contacted by telephone on 8201 3116, by fax on 8201 2035 or by email human.researchethics@flinders.edu.au

11.4 INTERVIEW INFORMATION SHEET

Title: Developing A Flexible and Adaptive Shipyard Fleet for Last Mile Delivery

Researchers:

Richard Ellis

College of Science and Engineering

Flinders University

Email: Richard.ellis@flinders.edu.au

Description of the study

As part of the Ship building program, The Defence Company is interested in implementing Autonomous Ground Vehicles (AGVs) to move components and materials to various locations in the shipyard. This is intended to not only increase cost efficiencies but also improve safety for workers by way of reduction of heavy lifting.

The study would provide modelling and intelligence to inform decision making within the defence company into fleet delivery strategy and specifically the number of vehicles required under various scenarios for effective performance.

Purpose of the study

The objective of this project is to explore the effective use of autonomous technologies to transport goods in shipbuilding as part of the warehouse and distribution strategy, that can involve autonomous ground vehicles, drones and forklifts.

What will I be asked to do?

You are invited to attend a one-on-one interview with a researcher who will ask you a few questions regarding your views about your:

- familiarity and experience with technologies,
- experiences of working in manufacturing environments
- expectations of the contributions of digital technologies to business outcomes,
- attitudes and beliefs about using technology, and
- ideas for how technology could be successfully implemented to support the workforce.

Participation is entirely voluntary, and you may withdraw at any stage without disadvantage to your relationship with Flinders University and its staff and students. The interview will take about 30 minutes and be conducted either by face-to-face or tele/video conference via computer at a time that is convenient for you.

With your consent, the interview will be audio recorded using a digital voice recorder to help with reviewing the results. Once recorded, the interview will be transcribed (typed-up) and stored as a computer file.

What benefit will I gain from being involved in this study?

Before the interview, you would be able a simulation on a computer of the modelling system including AGVs. This experience can give you an insight into a potential future application in shipbuilding. Also, sharing of your experiences and thoughts will contribute to identifying barriers and limitations that helps accelerate the uptake and diffusion of innovative technologies in shipbuilding in Australia.

Will I be identifiable by being involved in this study?

All responses will be confidential and anonymous and not identifiable to you. All information and results obtained in this study will be stored in a secure way, with access restricted to relevant researchers.

Are there any risks or discomforts if I am involved?

The researchers anticipate few risks from your involvement in this study beyond a time commitment. However, if the interview unintentionally triggers distress relating to workplace incidents, please contact your EAP provider or Lifeline on 13 11 14, free of charge to all participants. If you have any concerns regarding anticipated or actual risks or discomforts, please raise them with the researcher.

How do I agree to participate?

Participation is voluntary. You may answer 'no comment' or refuse to answer any questions, and you are free to withdraw from the interview at any time without effect or consequences. A consent form accompanies this information sheet. If you agree to participate please read and sign the form and send it back to us at richard.ellis@flinders.edu.au. Where this is not possible, verbal consent will be gained and recorded prior to the interview commencing.

How will I receive feedback?

A summary of the main findings of the study would be provided to all participants on conclusion of the study, via email.

Thank you for taking the time to read this information sheet, and we hope that you will accept our invitation to be involved.

This research project has been approved by the Flinders University Social and Behavioural Research Ethics Committee in South Australia (Project number INSERT PROJECT No. here following approval). For queries regarding the ethics approval of this project please contact the Executive Officer of the Committee via telephone on +61 8 8201 3116 or email human.researchethics@flinders.edu.au

11.5 INTERVIEW GUIDE

Preamble [read to participants after reading the information sheet and achieving consent]

As part of the Ship building program, Defence Company Australia is interested in implementing Autonomous Ground Vehicles (AGVs) and potentially aerial drones to move components and materials to various locations in the shipyard. This is intended to not only increase cost efficiencies but also improve safety for workers by way of reduction of heavy lifting.

Consequently, the objective of this project is to explore the effective use of autonomous technologies to transport goods in shipbuilding as part of the warehouse and distribution strategy, that can involve AGVs, drones and forklifts.

The study has developed a modelling, simulation of the number of vehicles required under various scenarios for effective performance.

This interview helps us to better understand your perspectives in the use of these technologies.

11.5.1 PERFORMANCE EXPECTANCY

Would the use of the technology assist employees with their performance?

11.5.2 EFFORT EXPECTANCY

Is the tool easy to use?

What would allow for simpler interaction with the simulation?

11.5.3 SAFETY / TRUST IN TECHNOLOGY

Do you have any safety concerns in the use of AGVs in the shipyard?

Do you trust the technology? Is it reliable?

In what situations would drones be useful? What would be the benefit? Dlivery of small consumables, unplanned small items quickly, stock taking? RFID reading?

Do you have any safety concerns in the use of drones in the shipyard?

Do you trust the technology?

11.5.4 CULTURE

What is the current culture to the use of simulation software as a major factor when making decisions?

Do you think there is a culture in shipbuilding to use this simulation technology?

11.5.5 OUTCOMES

On a scale of 1-5 (where 1 = no impact and 5= impact), to what extent would these technologies lead to an increase in

1. Safety
2. Accuracy
3. Productivity
4. Cost Efficiency
5. Innovation

Explain your selections.

11.5.6 INNOVATIVENESS

Do you think the use of modelling would lead to innovativeness in the shipyard?

How would you expect the simulation to evolve over time?

11.5.7 BEHAVIOURAL INTENTION

Would you use such as simulation system?

Yes/No. Why/ why not?

11.5.8 DEMOGRAPHIC

questions [survey can match to different technology types and level of awareness]

1. What is your age? – provide 5 year age bands? OR do we need 2 separate protocols for the 2 groups
2. What is your sex? M,F, other
3. What is your highest qualification?
4. How many years have your worked in shipbuilding?
5. What is your job role?
6. Were you born in Australia? Y, N, where?
7. Is English your first language?
8. Are you from an Aboriginal/Torres Strait Islander background? Y N

12 REFRENCES

'3D Simulation Modeling and Analysis Software' *FlexSim*, viewed 29 May 2021, <https://www.flexsim.com/>.

'4 Top Robotics Solutions Impacting The Shipbuilding Industry' 2019, *StartUs Insights*, viewed 24 January 2021, <https://www.startus-insights.com/innovators-guide/4-top-robotics-startups-impacting-the-shipbuilding-industry/>.

'6.4 Optimizing Vehicle Size and Fleet Size', viewed 9 February 2021, <https://brtguide.itdp.org/branch/master/guide/service-planning/optimizing-vehicle-size-and-fleet-size>.

'7 Big Problems with the Internet of Things' *CMSWire.com*, viewed 27 January 2021, <https://www.cmswire.com/cms/internet-of-things/7-big-problems-with-the-internet-of-things-024571.php>.

14:00-17:00 'ISO/TS 15066:2016', *ISO*, viewed 30 January 2021, <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/29/62996.html>.

'A* Search Algorithm' 2016, *GeeksforGeeks*.

'A* search algorithm' 2021, *Wikipedia*.

'Advantages and disadvantages of simulation - Computer simulation - GCSE ICT Revision' *BBC Bitesize*, viewed 28 January 2021, <https://www.bbc.co.uk/bitesize/guides/zvxp34j/revision/3>.

'AGVs drive warehouses of the future', viewed 7 April 2020, <https://www.dematic.com/en-au/downloads-and-resources/white-papers/agvs-drive-warehouses-of-the-future/>.

Andritsos, F & Perez-Prat, J 2000, 'The Automation and Integration of Production Processes in Shipbuilding', , p. 100.

‘AnyLogic: Simulation Modeling Software Tools & Solutions for Business’, viewed 29 May 2021, <https://www.anylogic.com/>.

‘Arena Simulation’, viewed 29 May 2021, <https://www.arenasimulation.com/>.

Arora, S, Raina, AK & Mittal, AK 2000, ‘Collision avoidance among AGVs at junctions’, in *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No.00TH8511)*, pp. 585–589.

Azimi, P, Haleh, H & Alidoost, M 2010, ‘The Selection of the Best Control Rule for a Multiple-Load AGV System Using Simulation and Fuzzy MADM in a Flexible Manufacturing System’, *Modelling and Simulation in Engineering*, vol. 2010, p. e821701.

Beamon, BM 1998, ‘Performance, reliability, and performability of material handling systems’, *International Journal of Production Research*, vol. 36, no. 2, pp. 377–393.

Becker, T & Stern, H 2016, ‘Future Trends in Human Work area Design for Cyber-Physical Production Systems’, *Procedia CIRP*, vol. 57, pp. 404–409.

Bilge, Ü & Tanchoco, JMA 1997, ‘AGV systems with multi-load carriers: Basic issues and potential benefits’, *Journal of Manufacturing Systems*, vol. 16, no. 3, pp. 159–174.

Bostelman, R 2009, ‘Towards improved forklift safety: white paper’, in *Proceedings of the 9th Workshop on Performance Metrics for Intelligent Systems*, PerMIS ’09, Association for Computing Machinery, Gaithersburg, Maryland, pp. 297–302.

Casini, M & Garulli, A 2016, ‘MARS: An Educational Environment for Multiagent Robot Simulations’, *Modelling and Simulation in Engineering*, vol. 2016, p. e5914706, viewed 14 February 2021, <https://www.hindawi.com/journals/mse/2016/5914706/>.

C-H, H & Egbelu, P 2000, ‘A framework for the selection of idle vehicle home locations in an automated guided vehicle system’, *International Journal of Production Research*, vol. 38, pp. 543–562.

Chemweno, P, Pintelon, L & Decre, W 2020, ‘Orienting safety assurance with outcomes of hazard analysis and risk assessment: A review of the ISO 15066 standard for collaborative robot systems’, *Safety Science*, vol. 129, p. 104832.

Chen, M-T (Mark), Mcginnis, L & Zhou, C 1999, ‘Design and operation of single-loop dual-rail inter-bay material handling system’, *International Journal of Production Research*, vol. 37, no. 10, pp. 2217–2237.

‘ch-en-manufacturing-industry-4-0-24102014.pdf’.

Chia-Han Lin & Ling-Ling Wang 1997, ‘Intelligent collision avoidance by fuzzy logic control’, *Robotics and Autonomous Systems*, vol. 20, no. 1, pp. 61–83.

Cimini, C, Lagorio, A, Pirola, F & Pinto, R 2019, ‘Exploring human factors in Logistics 4.0: empirical evidence from a case study’, *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 2183–2188.

‘Conflict-free shortest-time bidirectional AGV routeing: International Journal of Production Research: Vol 29, No 12’, viewed 18 February 2021, <https://www.tandfonline.com/doi/abs/10.1080/00207549108948090>.

‘Control and Simulate Multiple Warehouse Robots - MATLAB & Simulink’, viewed 15 February 2021, <https://www.mathworks.com/help/robotics/ug/control-and-simulate-multiple-warehouse-robots.html>.

‘Control and Simulate Multiple Warehouse Robots - MATLAB & Simulink - MathWorks Australia’, viewed 30 May 2021, <https://au.mathworks.com/help/robotics/ug/control-and-simulate-multiple-warehouse-robots.html>.

Davenport, TH, Barth, P & Bean, R ‘How “Big Data” Is Different’, , p. 5.

‘defence-briefing-note-oct17.pdf’.

Dewa, P, Pujawan, N & Vanany, I 2017, ‘Human errors in warehouse operations: an improvement model’, *International Journal of Logistics Systems and Management*, vol. 27, p. 298.

Digani, V, Hsieh, MA, Sabattini, L & Secchi, C 2019, ‘Coordination of multiple AGVs: a quadratic optimization method’, *Autonomous Robots*, vol. 43, no. 3, pp. 539–555.

Dijkhuizen, S 2017, ‘The history of the conveyor belt’, *Habasit Expert Blog*, viewed 19 January 2021, <http://blog.habasit.com/2017/12/history-conveyor-belt/>.

Djuric, AM, Urbanic, RJ & Rickli, JL 2016, ‘A Framework for Collaborative Robot (CoBot) Integration in Advanced Manufacturing Systems’, *SAE International Journal of Materials and Manufacturing*, vol. 9, no. 2, pp. 457–464.

Ebben, M 2001, ‘Logistic control in automated transportation networks’,.

Egbelu, PJ 1987, ‘The use of non-simulation approaches in estimating vehicle requirements in an automated guided based transport system’, *undefined*, viewed 9 February 2021, </paper/The-use-of-non-simulation-approaches-in-estimating-Egbelu/f6706a695a8382ab5c6dbddf9a03fba3430d632f>.

Egbelu, PJ 1991, ‘Reduction of manufacturing lead-time through selection of machining rate and unit handling size’, *Engineering Costs and Production Economics*, vol. 21, no. 1, pp. 21–34.

Egbelu, PJ 1993, ‘Positioning of automated guided vehicles in a loop layout to improve response time’, *European Journal of Operational Research*, vol. 71, no. 1, pp. 32–44.

Fantini, P, Tavola, G, Taisch, M, Barbosa, J, Leitao, P, Liu, Y, Sayed, MS & Lohse, N 2016, ‘Exploring the integration of the human as a flexibility factor in CPS enabled manufacturing environments: Methodology and results’, in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 5711–5716.

FERGAL, G 2018, ‘Conveyor system disadvantages and alternatives’, *6 River Systems*.

Fethi, A & Mehdi, S 2019, 'The effect of AGVs number on a flexible manufacturing system', in *2019 International Conference on Applied Automation and Industrial Diagnostics (ICAAID)*, pp. 1–5.

Fletcher, SR, Johnson, T, Adlon, T, Larreina, J, Casla, P, Parigot, L, Alfaro, PJ & Otero, M del M 2020, 'Adaptive automation assembly: Identifying system requirements for technical efficiency and worker satisfaction', *Computers & Industrial Engineering*, vol. 139, p. 105772.

'Forklifts vs. Automated Guided Vehicles' *Adaptalift*, viewed 20 January 2021, <https://www.adaptalift.com.au/blog/2013-07-15-forklifts-vs-automated-guided-vehicles>.

GASKINS, RJ & TANCHOCO, JMA 1987, 'Flow path design for automated guided vehicle systems', *International Journal of Production Research*, vol. 25, no. 5, pp. 667–676.

Gorecky, D, Schmitt, M, Loskyll, M & Zühlke, D 2014, 'Human-machine-interaction in the industry 4.0 era', in *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*, pp. 289–294.

Grahn, S, Johansen, K & Eriksson, Y 2017, 'Safety Assessment Strategy for Collaborative Robot Installations', *Robots Operating in Hazardous Environments*.

Grosse, EH, Calzavara, M, Glock, CH & Sgarbossa, F 2017, 'Incorporating human factors into decision support models for production and logistics: current state of research', *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6900–6905.

Guo, X, Yu, Y & Koster, RBMD 2016, 'Impact of required storage space on storage policy performance in a unit-load warehouse', *International Journal of Production Research*, vol. 54, no. 8, pp. 2405–2418.

Hamdy, A *Optimization of Automated Guided Vehicles (AGV) Fleet Size With Incorporation of Battery Management*, Old Dominion University Libraries.

Hecklau, F, Galeitzke, M, Flachs, S & Kohl, H 2016, 'Holistic Approach for Human Resource Management in Industry 4.0', *Procedia CIRP*, vol. 54, pp. 1–6.

‘How Do Conveyor Belts Work? | Belt Functions, Uses & Applications’ 2019, *SEMCOR*, viewed 20 January 2021, <https://www.semcor.net/blog/how-do-conveyor-belts-work/>.

‘Improving safety and efficiency of AGVs at warehouse black spots - IEEE Conference Publication’, viewed 17 February 2021, <https://ieeexplore.ieee.org/abstract/document/6937004?casa_token=VNgQUUdMw4wAAAAA:ofiNC0q5OxAZLpdq1XJSFf61RBSG1Pd1TM9PfE8leE8-KYRdGgsJQolGo3GWlU7P9MeIZ7K_lvta2QU>.

‘innovativeness’ *The Free Dictionary*.

‘Introduction to A*’, viewed 27 May 2021, <http://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html>.

Johnson, ME 2001, ‘Modelling empty vehicle traffic in AGVS design’, *International Journal of Production Research*, vol. 39, no. 12, pp. 2615–2633.

Kadir, BA, Broberg, O & Conceição, CS da 2019, ‘Current research and future perspectives on human factors and ergonomics in Industry 4.0’, *Computers & Industrial Engineering*, vol. 137, p. 106004.

Kaspi, M, Kesselman, U & Tanchoco, JMA 2002, ‘Optimal solution for the flow path design problem of a balanced unidirectional AGV system’, *International Journal of Production Research*, vol. 40, no. 2, pp. 389–401.

KASPI, M & TANCHOCO, JMA 1990, ‘Optimal flow path design of unidirectional AGV systems’, *International Journal of Production Research*, vol. 28, no. 6, pp. 1023–1030.

Kiel, D, Müller, JM, Arnold, C & Voigt, K-I 2017, ‘Sustainable industrial value creation: benefits and challenges of industry 4.0’, *International Journal of Innovation Management*, vol. 21, no. 08, p. 1740015.

KIM, CW & TANCHOCOJ, JMA 1993, ‘Operational control of a bidirectional automated guided vehicle system’, *International Journal of Production Research*, vol. 31, no. 9, pp. 2123–2138.

Klumpp, M 2018, ‘Automation and artificial intelligence in business logistics systems: human reactions and collaboration requirements’, *International Journal of Logistics Research and Applications*, vol. 21, no. 3, pp. 224–242.

Korman, R 2019, ‘What is “safe enough” for drone deliveries?’, *The Seattle Times*, viewed 22 January 2021, <https://www.seattletimes.com/business/boeing-aerospace/what-is-safe-enough-for-drone-deliveries/>.

Koster, RBMD, Johnson, AL & Roy, D 2017, ‘Warehouse design and management’, *International Journal of Production Research*, vol. 55, no. 21, pp. 6327–6330.

Krnjak, A, Draganjac, I, Bogdan, S, Petrović, T, Miklić, D & Kovačić, Z 2015, ‘Decentralized control of free ranging AGVs in warehouse environments’, in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2034–2041.

Kuo, P-H, Krishnamurthy, A & Malmborg, CJ 2007, ‘Design models for unit load storage and retrieval systems using autonomous vehicle technology and resource conserving storage and dwell point policies’, *Applied Mathematical Modelling*, vol. 31, no. 10, pp. 2332–2346.

Le-Anh, T & de Koster, MBM 2004, *A Review of Design and Control of Automated Guided Vehicle Systems*, ID 594969, Social Science Research Network, Rochester, NY, viewed 18 February 2021, <https://papers.ssrn.com/abstract=594969>.

Lei, Y, Liu, J, Ni, J & Lee, J 2010, ‘Production line simulation using STPN for maintenance scheduling’, *Journal of Intelligent Manufacturing*, vol. 21, pp. 213–221.

Leite, L, Esposito, R, Vieira, A & Lima, F 2015, ‘Simulation of a Production Line with Automated Guided Vehicle: A Case Study’, *Independent Journal of Management & Production*, vol. 6.

Leopold, H, van der Aa, H & Reijers, HA 2018, ‘Identifying Candidate Tasks for Robotic Process Automation in Textual Process Descriptions’, in J Gulden, I Reinhartz-Berger, R Schmidt, S Guerreiro, W Guédria, & P Bera (eds), *Enterprise, Business-Process and Information Systems*

Modeling, Lecture Notes in Business Information Processing, Springer International Publishing, Cham, pp. 67–81.

Lim, J, Lim, J, Yoshimoto, K, Kim, K & Takahashi, T 2002, ‘A construction algorithm for designing guide paths of automated guided vehicle systems’, *International Journal of Production Research - INT J PROD RES*, vol. 40, pp. 3981–3994.

Lyu, X, Song, Y, He, C, Lei, Q & Guo, W 2019a, ‘Approach to Integrated Scheduling Problems Considering Optimal Number of Automated Guided Vehicles and Conflict-Free Routing in Flexible Manufacturing Systems’, *IEEE Access*, vol. 7, pp. 74909–74924.

Lyu, X, Song, Y, He, C, Lei, Q & Guo, W 2019b, ‘Approach to Integrated Scheduling Problems Considering Optimal Number of Automated Guided Vehicles and Conflict-Free Routing in Flexible Manufacturing Systems’, *IEEE Access*, vol. 7, pp. 74909–74924.

MAHADEVAN, B & NARENDRAN, TT 1990, ‘Design of an automated guided vehicle-based material handling system for a flexible manufacturing system’, *International Journal of Production Research*, vol. 28, no. 9, pp. 1611–1622.

Mahadevan, B & Narendran, TT 1992, ‘Determination of unit load sizes in an AGV-based material handling system for an FMS’, *International Journal of Production Research*, vol. 30, no. 4, pp. 909–922.

Mahulea, C & Kloetzer, M 2015, ‘Planning mobile robots with Boolean-based specifications’, *Proceedings of the IEEE Conference on Decision and Control*, vol. 2015, pp. 5137–5142.

‘Maintenance Theory of Reliability, Springer Series in Reliability Engineering by Toshio Nakagawa | 9781852339395 | Booktopia’, viewed 17 February 2021, <https://www.booktopia.com.au/maintenance-theory-of-reliability-toshio-nakagawa/book/9781852339395.html>.

‘MARS - Multi-Agent Robot Simulator’, viewed 14 February 2021, <http://mars.diism.unisi.it/>.

‘Mobile Robot Algorithm Design - MATLAB & Simulink’, viewed 14 February 2021, <https://www.mathworks.com/help/robotics/ground-vehicle-algorithms.html>.

‘Mobile Robots and Drones in Material Handling and Logistics 2018-2038: IDTechEx’, viewed 21 January 2021, <https://www.idtechex.com/en/research-report/mobile-robots-and-drones-in-material-handling-and-logistics-2018-2038/548>.

Moktadir, MdA, Ali, SM, Kusi-Sarpong, S & Shaikh, MdAA 2018, ‘Assessing challenges for implementing Industry 4.0: Implications for process safety and environmental protection’, *Process Safety and Environmental Protection*, vol. 117, pp. 730–741.

Mourtzis, D, Doukas, M & Bernidaki, D 2014, ‘Simulation in Manufacturing: Review and Challenges’, *Procedia CIRP*, vol. 25, pp. 213–229.

Mourtzis, D, Papakostas, N, Mavrikios, D, Makris, S & Alexopoulos, K 2015, ‘The role of simulation in digital manufacturing: applications and outlook’, *International Journal of Computer Integrated Manufacturing*, vol. 28, no. 1, pp. 3–24.

Mourtzis, D, Vlachou, E & Milas, N 2016, ‘Industrial Big Data as a Result of IoT Adoption in Manufacturing’, *Procedia CIRP*, vol. 55, pp. 290–295.

Mousavi, M, Yap, HJ, Musa, SN, Tahriri, F & Dawal, SZM 2017, ‘Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization’, *PLOS ONE*, vol. 12, no. 3, p. e0169817.

Moussa, A 2020, ‘Manufacturing 4.0 Operations Scheduling with AGV Battery Management Constraints’,.

Németh, I, Püspöki, J, Viharos, AB, Zsóka, L & Pirka, B 2019, ‘Layout configuration, maintenance planning and simulation of AGV based robotic assembly systems’, *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 1626–1631.

Nuzzi, C, Pasinetti, S, Lancini, M, Docchio, F & Sansoni, G 2019, 'Deep learning-based hand gesture recognition for collaborative robots', *IEEE Instrumentation Measurement Magazine*, vol. 22, no. 2, pp. 44–51.

O'Connor, PDT 1994, 'Reliability and risk assessment, J. D. Andrews and T. R. Moss, Longman Scientific and Technical, 1993. Number of pages: 368. Price: £48.00', *Quality and Reliability Engineering International*, vol. 10, no. 3, pp. 251–251.

Pearce, M, Mutlu, B, Shah, J & Radwin, R 2018, 'Optimizing Makespan and Ergonomics in Integrating Collaborative Robots Into Manufacturing Processes', *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1772–1784.

Peters, B & Yang, T 1997, 'Integrated facility layout and material handling system design in semiconductor fabrication facilities',.

Qi, M, Li, X, Yan, X & Zhang, C 2018, 'On the evaluation of AGVS-based warehouse operation performance', *Simulation Modelling Practice and Theory*, vol. 87, pp. 379–394.

Rajotia, S, Shanker, K & Batra, JL 1998, 'Determination of optimal AGV fleet size for an FMS', *International Journal of Production Research*, vol. 36, no. 5, pp. 1177–1198.

Ramirez-Peña, M, Sánchez Sotano, AJ, Pérez-Fernandez, V, Abad, FJ & Batista, M 2020, 'Achieving a sustainable shipbuilding supply chain under I4.0 perspective', *Journal of Cleaner Production*, vol. 244, p. 118789.

Ren, K, Thomson, A & Abadi, DJ 2014, 'An evaluation of the advantages and disadvantages of deterministic database systems', *Proceedings of the VLDB Endowment*, vol. 7, no. 10, pp. 821–832.

Riazi, S, Bengtsson, K & Lennartson, B 2020, 'Energy Optimization of Large-Scale AGV Systems', *IEEE Transactions on Automation Science and Engineering*, pp. 1–12.

Rodič, B 2017, 'Industry 4.0 and the New Simulation Modelling Paradigm', *Organizacija*, vol. 50, no. 3, pp. 193–207.

Romaine, E 2018, ‘Alternatives to Automated Guided Vehicles (AGVs)’, *Conveyco*.

Sagiroglu, S & Sinanc, D 2013, ‘Big data: A review’, in *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pp. 42–47.

Schneider, R ‘Robotic Automation Can Cut Costs’, , p. 3.

Schrauf, RG, Jesper Vedsø, and Stefan ‘A strategist’s guide to Industry 4.0’, *strategy+business*, viewed 26 January 2021, <https://www.strategy-business.com/article/A-Strategists-Guide-to-Industry-4.0?gko=a2260>.

Sedehi, MS & Farahani, RZ 2009, ‘An integrated approach to determine the block layout, AGV flow path and the location of pick-up/delivery points in single-loop systems’, *International Journal of Production Research*, vol. 47, no. 11, pp. 3041–3061.

Sezer, OB, Dogdu, E & Ozbayoglu, AM 2018, ‘Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey’, *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 1–27.

Shafique, K, Khawaja, BA, Sabir, F, Qazi, S & Mustaqim, M 2020, ‘Internet of Things (IoT) for Next-Generation Smart Systems: A Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios’, *IEEE Access*, vol. 8, pp. 23022–23040.

Shavarani, SM, Nejad, MG, Rismanchian, F & Izbirak, G 2018, ‘Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of Amazon prime air in the city of San Francisco’, *The International Journal of Advanced Manufacturing Technology*, vol. 95, no. 9, pp. 3141–3153.

Shi, H, Yang, M & Jiang, P 2021, ‘Social Production System: A Three-layer Smart Framework for Implementing Autonomous Human-machine Collaborations in A Shop Floor’, *IEEE Access*, pp. 1–1.

Sisinni, E, Saifullah, A, Han, S, Jennehag, U & Gidlund, M 2018, ‘Industrial Internet of Things: Challenges, Opportunities, and Directions’, *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734.

Srinivasan, M, Bozer, Y & Cho, M 1994, 'Trip-based material handling systems: throughput capacity analysis', *IIE Transactions*, vol. 26, pp. 70–89.

Stokes, P 2019, '4 Stages of IoT architecture explained in simple words', *Medium*, viewed 27 January 2021, <https://medium.com/datadriveninvestor/4-stages-of-iot-architecture-explained-in-simple-words-b2ea8b4f777f>.

Talbot, L 2003, *Design and performance analysis of multistation automated guided vehicle systems*, Louvain-la-Neuve : CIACO.

Tanchoco, JMA & Co, CG 1994, 'Real-time control strategies for multiple-load AGVs', in JMA Tanchoco (ed.), *Material Flow Systems in Manufacturing*, Springer US, Boston, MA, pp. 300–331.

Tao Yifei, Chen Junruo, Liu Meihong, Liu Xianxi, & Fu Yali 2010, 'An estimate and simulation approach to determining the Automated Guided Vehicle fleet size in FMS', in *2010 3rd International Conference on Computer Science and Information Technology*, pp. 432–435.

Tavana, M, Fazlollahtabar, H & R., H 2014, 'A Bi-Objective Stochastic Programming Model for Optimising Automated Material Handling Systems with Reliability Considerations', *International Journal of Production Research*, vol. 52, pp. 5597–5610.

Tjahjono, B, Esplugues, C, Ares, E & Pelaez, G 2017, 'What does Industry 4.0 mean to Supply Chain?', *Procedia Manufacturing*, vol. 13, pp. 1175–1182.

Trenkle, A, Seibold, Z & Stoll, T 2013, 'Safety requirements and safety functions for decentralized controlled autonomous systems', in *2013 XXIV International Conference on Information, Communication and Automation Technologies (ICAT)*, pp. 1–6.

turvoinc 2017, 'Warehouse Automation: Warehouse Workers Need Not Fear Automation', *Transportation Management Company | Cerasis*.

V G, B & Patil, R 2015, *Virtual Manufacturing: A Review*,.

Valmiki, P, Simha Reddy, A, Panchakarla, G, Kumar, K, Purohit, R & Suhane, A 2018, ‘A Study on Simulation Methods for AGV Fleet Size Estimation in a Flexible Manufacturing System’, *Materials Today: Proceedings*, vol. 5, no. 2, Part 1, pp. 3994–3999.

Venkatesh, V, Morris, MG, Davis, GB & Davis, FD 2003, ‘User Acceptance of Information Technology: Toward a Unified View’, *MIS Quarterly*, vol. 27, no. 3, pp. 425–478.

Vis, IFA 2006, ‘Survey of research in the design and control of automated guided vehicle systems’, *European Journal of Operational Research*, vol. 170, no. 3, pp. 677–709.

Vivaldini, K, Rocha, L, Fróes, N, Becker, M & Moreira, A 2015, ‘Integrated tasks assignment and routing for the estimation of the optimal number of AGVS’, *The International Journal of Advanced Manufacturing Technology*, vol. 82, pp. 1–18.

Vysocky, A & Novak, P 2016, ‘Human - Robot collaboration in industry’, *MM Science Journal*, vol. 2016, pp. 903–906.

Watson, N 2021, ‘AGV Numbers and Charging stations’,.

‘What does unit load mean?’, viewed 4 February 2021, <https://www.definitions.net/definition/unit+load>.

‘What is the Knapsack Problem? - Definition from Techopedia’ *Techopedia.com*, viewed 27 October 2021, <http://www.techopedia.com/definition/20272/knapsack-problem>.

‘When-it-comes-to-powering-your-AGVs-whitepaper-FINAL.pdf’.

Wu, N & Zhou, M 2007, ‘Shortest Routing of Bidirectional Automated Guided Vehicles Avoiding Deadlock and Blocking’, *Mechatronics, IEEE/ASME Transactions on*, vol. 12, pp. 63–72.

Yan, R, Dunnett, SJ & Jackson, LM 2018, ‘Novel methodology for optimising the design, operation and maintenance of a multi-AGV system’, *Reliability Engineering & System Safety*, vol. 178, pp. 130–139.

Yan, X, Zhang, C & Qi, M 2017a, ‘Multi-AGVs Collision-Avoidance and Deadlock-Control for Item-To-Human Automated Warehouse’, in *2017 International Conference on Industrial Engineering, Management Science and Application (ICIMSA)*, pp. 1–5.

Yan, X, Zhang, C & Qi, M 2017b, ‘Multi-AGVs Collision-Avoidance and Deadlock-Control for Item-To-Human Automated Warehouse’, in *2017 International Conference on Industrial Engineering, Management Science and Application (ICIMSA)*, pp. 1–5.

Złotowski, J ‘building positive environment for humans and robots - Google Scholar’, viewed 1 February 2021, <https://scholar.google.com/scholar?hl=en&as_sdt=0,5&q=building+positive+environment+for+humans+and+robots>.