ALMA MATER STUDIORUM - UNIVERSITA' DI BOLOGNA

Corso di Laurea Magistrale in Ingegneria Gestionale

FACTORS AFFECTING THE MISSION TIME OF AN AUTONOMOUS MOBILE ROBOT: A THEORETICAL AND EXPERIMENTAL INVESTIGATION

Tesi di laurea in

Sistemi di Produzione Avanzati

Relatore: Prof. Cristina Mora

Correlatore: Prof. Fabio Sgarbossa
Correlatore: Prof. Mirco Peron

Presentata da: Federico Castelluzzo

Appello
Primo.

Anno accademico
2021-2022

ACKNOWLEDGEMETS

First things first, I would like to thank Professor Fabio Sgarbossa for his support, patience and for letting me in this exchange program at the Norwegian University of Science and Technology, which gave me the opportunity to discover a new university, a new country and a new culture. Then I would like to thank research fellow Mirco Peron for his extreme patience, for following me daily and repeatedly explaining things to me. He’s always been available, even more than he should have and his support was key to this work.

Then I want to thank all the people I’ve met during this exchange, from my flat mates to my closest friends, they’ve all been fundamental in this period, helping me going through hard times and keeping my level of concentration high.

Last but not least, I’d like to thank my family, especially my parents, for always believing in me, for allowing me to come to achieve this experience and for sticking by my side throughout my whole university career.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Industry 4.0 has already landed in almost every developed country in the world. Its advantages are so clear and evident that they have turned the manufacturing world upside down. But what exactly are we talking about when we talk about Industry 4.0? What are its paradigms, its key points and its methods to implement all the changes we are determined to bring to the table?
Industry 4.0 is discussed as the new border of manufacturing as it's based on ideas and principles that follow the many current steps forward made by information systems and technologies. Therefore, the keys to this new manufacturing approach are horizontal and vertical integration, digitalization, automation.
The development brought by Industry 4.0 applies to many areas of the supply chain, from information flows to operations and logistics. This last field is particularly important since it's considered a key domain for examining a successful Industry 4.0 implementation. In logistics and intralogistics, the highest priority is given to developing material handling systems that can provide a high level of flexibility, automation and responsiveness to dynamic changes. The ideal model is thought as an autonomous one, in which each transport unit is not ruled by a central unit of command but, on the contrary, is part of a decentralized system of material handlers that can communicate with each other thanks to their high connectivity and ability to collect data and rapidly exchange it with the company cloud. As predictable, all of these targets wouldn't be reached with traditional AGV-based material handling system, which happens to be too rigid and centralized to be effective. This work puts another kind of devices in the spotlight, a more flexible one, a smarter one and a more unsupervised one: Autonomous Mobile Robots (AMR).
Robots like these are able to freely move in an environment interacting with it by mapping it and rapidly reacting to every change that's brought to it by external actors like operators or other robots. Unlike AGVs, there's no need for a predetermined path, which can require a significant amount of time and resources to be implemented.
These devices are characterized by autonomy of decision and freedom to move in the surrounding space in order to react to unexpected events such as the sudden showing up of an obstacle. This is a major advantage but, at the same time, it may affect the time a robot takes to complete a mission. Such loss, in terms of time or equivalent distance, has its origin in a situation that can normally happen in a working environment and therefore must be taken into consideration during the estimation

of the mission time of an AMR and, consequently, during the organization of the plant layout or the dimensioning of the fleet needed to manage a material handling system in such productive plant. Decisions like the one which has just been outlined are of great relevance to the company since they can imply the spending or saving of a significant amount of money.

In this work the multiple advantages brought by AMRs will be shortly introduced and discussed with a literature review and then the focus will be moved on to analyzing how robots behave in certain conditions of the surrounding environment, going deep into details about how much time is lost due to each obstacle, each change of direction, each turning and so on. To do so, it will be necessary to have a clear idea of how AMRs work, which will be gained by multiple experiences in NTNU logistic 4.0 laboratory. In the meantime, a logistics simulation software, AnyLogic, will be used to reproduce and simulate each scenario that's considered relevant and coherent with reality to be eventually studied and modeled.

Thanks to the simulations and experiences in the laboratory, several parameters will be detected according to the way they affect the mission time of an Autonomous Mobile Robot. For each of those parameters, and for all the combinations of them, will be generated a function that establishes a relation between the variation of such parameter and the mission time variation. To do so, it will be necessary using some data analysis programs and applications such as Minitab and MatLab.
This is how the goal of this thesis will be reached, by building a model that couples a certain amount of mission time variation to each possible scenario reproduced through AnyLogic simulations.

EXECUTIVE SUMMARY

This thesis is structured as following. The first part will be an introduction of what AMRs are, what their role in modern industry is and what main advantages they bring in response to the need to fit in the new boundaries of Industry 4.0. This is made through a deep review of the existing literature about these topics, which enables to have the right perspective of the key role autonomous robots play in modern intralogistics, understanding why companies choose them and why they turn out to be this successful. At the end of this section a short description of the case study of this thesis is reported, along with a brief summary about AnyLogic simulation software and its structure.

In the following section all the parameters that affect the mission time are pointed out and, for each of them, a deep reflection leads to creating the simulation environment, running the program and analyzing the results, which will be then graphed and furtherly discussed in the following section.

As anticipated, in this next section the results of the previous simulations are compared, discussed and graphed through a program called MatLab. For each of them, a function that associates each parameter variation to a mission time variation.

The last part of this work is strictly connected to the previous because the focus is on creating new scenarios with the combination of all the parameters previously detected. This is followed by a final analysis of the variance, led using Minitab statistic application, which will give as output useful information about which parameters are more significant in terms of influence on the mission time. Further investigations and final conclusions are the closing part of this work.

1. INTRODUCTION AND OBJECTIVES

Up to the present, the world has been through four industry revolutions and, among all four of those, the latter industry revolution that occurred in the more recent past has revealed new potentials and brought forward the catchword digitalization. This is one of the terms on which the whole idea of Industry 4.0 is centered on. When we talk about digitalization we mean a change which is more structural than just digitalizing existing processes or products. It has been defined as *the use of digital technologies to change a business model and provide new revenue and new value-adding opportunities. It's the process of moving to a digital business.*

This definition, given by Gartner in 2019, implies that digitalization doesn't only involve manufacturing processes and the way they are brought on; on the contrary it involves every single component of the supply chain, in order to create a web of business processes integrated and synchronized to work as a whole unit. In order to reach this target logistics and intralogistics are a key domain because of its function of providing inputs for production systems and harmonizing the whole supply chain. Since logistics is involved in a wide range of activities, there are multiple opportunities to improve a company performance and increase the potential of its value-adding activities through improvements in logistics. Therefore, new technologies and smart solutions have the power of immediately gaining strategic importance. For instance, big data and cloud operating, whose aims are respectively to form production networks capable of dynamic reconfiguration and high flexibility and providing global feedback to achieve high efficiency, represent powerful instruments and are considered as two of the main pillars of Industry 4.0.

Through Industry 4.0 connectivity, automation, fast information exchange and analytics, a new dimension of flexibility can be reached and new approaches to planning and controlling production systems designed.

This study focuses on how the previously mentioned key points are implemented in intralogistics to create a more flexible and cost-effective environment through introducing Autonomous Mobile Robots. Such devices represent an evolution of material handling systems because, compared to standard AGVs, which were considered the most advanced technology of the recent past years, they have some strategic and distinguishing characteristics. First, they don't need any predetermined path to move, since they are able to calculate what the best route is to get from a random point A to another point B. This is made possible by the high computational power of AMR system and is a fundamental point because it significantly decreases

the amount of time and resources spent to engineer and install path guides needed by AGVs, which required either an optical or a magnetic or a laser guiding system.
Secondly, AMRs are not managed by a central unit of control. On the contrary, thanks to the improvements like real-time algorithms made possible by Artificial Intelligence, they have an impressive autonomy of decision, such as dynamic routing and scheduling, which enables the whole material handling system to be way more decentralized and flexible. Another benefit that comes from adopting AMRs instead of more traditional methods is their ability to gain information about the environment around them in order to react to any unexpected event such as the presence of an obstacle in the path, which can be easily avoided by the robot, or the need to work in a new manufacturing cell where the robot might have never been before. Its adaptability to work under all circumstances is key to the material handling system flexibility and responsiveness.
The focus of this thesis is on the aspect that was mentioned in the previous lines: how robots interact with the work environment they must serve and how they react to unexpected events like an obstacle of various dimensions, speed and direction or the need to change direction multiple times to get to destination. The reason why observing these scenarios is fundamental is because, even though obstacle avoidance is computed with maximum precision by the processing unit of the device, a time loss is inevitable and its implications have not been object of any literature deepening.
It is therefore necessary to understand under which conditions and in what way the mission time of AMRs are affected. Detecting all the key factors that imply a negative variation on the performances of these devices is then the starting point of this work, for which a punctilious literature review is necessary. From this point on, the aim is parameterizing every factor that affects the mission time, in order to create a tool that associates a time loss function to each of the scenarios. The output of these activities will provide useful information for managers and engineers who have the task to develop a material handling system from scratch for a new production plant or to redesign an existing one. Indeed, time variations coming from our study are critical in the fleet dimensioning process or in a plant layout redesigning. Indeed, when a new productive plant must be designed, it is necessary to define the number of vehicles that must be introduced in the system in order to correctly fulfill the need of material of the whole productive plant. The output of this work helps in this process by allowing the engineer to consider the equivalent distance traveled by each robot rather than the mere distance between starting point and arrival point. The aspect just outlined might determine the need for more vehicles in the AMR fleet.

From the perspective of the urgency to execute a plant layout redesigning, this work highlights all the circumstances under which robots are affected in their performance and therefore provides engineers of suggestions about what layout bottlenecks and inefficiencies should be avoided in order to minimize the mission time loss.
As a matter of fact, the general aim of this work is maximizing the efficiency of a material handling system by considering the characteristics of robots' behavior, which is a lacking topic in literature.

To summarize, this work, in order to achieve the objectives that have just been pointed out, is structured to answer three key questions, which are the following.

- What are the situations that most affect the efficiency of AMRs in a material handling system?
- Out of those, which ones can be represented by parameters whose variations can be simulated and computed into a function?
- How much does each parameter affect the mission time of AMRs? How much a circumstance given by a combination of those parameters affects the mission time?

It is now important to clarify how these targets are accomplished, which sources are examined and which tools are used in the simulation and data analysis sections.

1.1 METHODOLOGY

To answer the key questions of this thesis, following their order, it is compulsory to carry out a detailed literature review to fully understand all the possible features and variants of Autonomous Mobile Robots. Since these devices can be adopted in several fields of human activity, their job is not standardized and so are their settings. Examining articles, producers' instructions or presentations and users' reviews can be beyond helpful. A classic approach has been used: renown academic website Scopus was the main source, as it provides the user of thousands of articles, book chapters and any kind of research publication as well as reliable data, metrics and analytical tools. Other similar websites such as GoogleScholar or ScienceDirect but also several unofficial websites were used in the making of this thesis to get a well-rounded knowledge of what robots are used for, what their advantages and disadvantages are and how they behave in every circumstance of pressure or uncomfortable environment. In addition to this, it was necessary to operate in the Logistics 4.0 laboratory, made available by NTNU to accomplish this study. Experiencing the lab made it possible to outline all the parameters that were to be set in the simulation software AnyLogic, copying those given by default to the AMRs available at the NTNU campus. The speed of the robot, the minimum distance to obstacles, speed in the nearing of an obstacle and many more were set up in the simulation environment after observing and measuring them during the laboratory tests.

Moreover, several real-life situations, which would have been difficult to run with AnyLogic, were reproduced in the lab and it was possible to notice how every now and then a circumstance under which a robot performs differently from the simulation software occurred. For instance, using AnyLogic it's possible for an AMR to get through a 60-centimeter aisle while in the lab it was possible to find out that these devices need at least 95 centimeters to get through.

As mentioned, software AnyLogic covered most of the time spent in the making of this project because it is used to reproduce every single scenario, starting from the base case, which will be discussed further in the work, to the ones coming from the combination of all parameters. Concisely, what was simulated through this software was a transportation mission of an item by an AMR under many different circumstances and characteristics of the surrounding working environment, obtained by changing the settings of some parameters or introducing obstacles or turns, and in particular its route from point A to point B whose distance remained unchanged in every simulation in order to make possible comparisons between each scenario and the base case. Basically, this approach is comparable to observing the flow of a liquid inside of a pipe. Indeed, just like in fluid mechanics, there will be head losses due to hostile conditions of the environment. Head losses are decreases of fluid pressure due to the friction between the moving fluid and the stationary pipe. In particular, this work is focused on minor losses, defined as energy losses localized in a specific component of the piping system like bends and valves, which can be logically translated into a material handling system as curves, obstacles or a narrowing of an aisle.

All the most important features and settings of AnyLogic will be examined in the next chapters.

Microsoft Excel is used to take count of all the simulation results, the time differences between such results and the base scenario and to create all the possible combinations of the chosen values of each parameter.

The data analysis is led mainly by using two applications, one for the computation of the mission time variation function for each parameter and one for the analysis of variance, a statistic analysis that will be investigated in the last chapter of this work.
For the first goal the application MATLAB and in particular its Curve Fitting tool is used. It consists of a programming and numeric computing platform used by millions of engineers and scientists to analyze data, develop algorithms, and create models.
The second goal is achieved by operating with Minitab, another advanced and powerful tool for statistics used for the analysis and optimization of business processes.
At the end of this work, it will be possible to extract a function that will allow to quantify the relation existing between each scenario, determined by one parameter or the combination of more, and the mission time loss that the above-mentioned scenario implies. What has just been cleared out implies that, since a scenario can be described by the variation of multiple parameters at the same time, the final formula is going to be a sum of components.

1.2 LIMITATIONS

There are some limitations that occurred in the making of this study which contributed to increasing the difficulty of more than one phase of it.
The main limitation come from the simulation software AnyLogic, with which there was very little familiarity. Therefore, it was difficult to try and learn its dynamics from scratch and gaining knowledge about it subtracted much time from the core activities of this thesis. Moreover, this software includes robots that work in a highly similar way to AMRs but there are some circumstances under which they perform differently. This will be discussed later in this work and will allow to understand which scenarios were not representable with AnyLogic. This aspect is particularly relevant because it leaves some open questions and future developments that could be easily investigated by a deeper use of robots in labs or production plants rather than in a simulation software.
One last limitation comes from the fact that the functions obtained by data analysis applications were approximated and, therefore, the results have a small percentage of error and uncertainty.

2. FEATURES AND APPLICATION FIELDS OF AUTONOMOUS MOBILE ROBOTS

In this section of the thesis a deepening of the main features of Autonomous Mobile Robots is executed, starting from describing the most relevant devices implanted on them and some of the most likely fields where they can be introduced.
To go into further details about the technical specifications of AMRs, the ones available at NTNU Logistics 4.0 are brought in the spotlight. The above-mentioned devices are MiR200, showed in Figure 1, whose name comes from the producer MiR, a leading manufacturer of collaborative mobile robots that was a first mover in this field. For this reason, their robots have experienced a wide spreading among the manufacturing field.
The reason of this expansion lies in its characteristics of simplicity of use and agility, but also its weight capability, as this robot can transport loads up to 200 kilograms. Its maximum speed is 1,1 m/s while going forwards but is generally set to slightly lower values.

Figure 1. A MiR200 Autonomous Mobile Robot

Since its dimensions are relatively small and the minimum width required to let the robot pass through a tight aisle or door, it can be used for multiple goals and situations. Its responsiveness to the surrounding environment is given by its cameras and sensors that can create a map of the plant they shall work in and a continuous activity of obstacle detection. The process through which a map of the plant is made takes place at the beginning of the robots' work. In this phase, the AMR is led manually around the plant so that its sensors can provide the central unit of information regarding the position of walls, workstations and any other kind of fixed element that might interfere with the trajectories traveled by robots. These obstacles are shown with black marks on the computer interface. Once this process is concluded the AMR can operate in full autonomy inside of the environment that was just mapped. Meanwhile, sensors keep detecting obstacle in order to prevent collisions with something that hadn't been revealed in the mapping process. In case something unexpected showed up, it would appear in the interface as red mark of the same shape as the real-life obstacle. In figure 2 an example of mapping output is shown. The just mentioned aspect is what contributes to robots' outstanding ability of obstacle avoidance. The robot-user interface can be easily mastered by anyone in a short amount of time and this is one further reason why AMRs find that much appreciation among manufacturing companies.

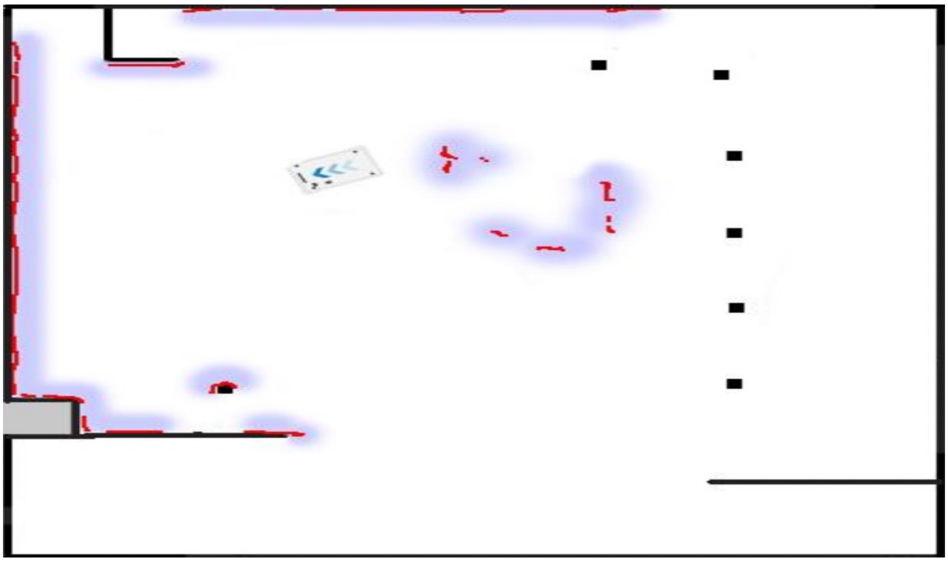


Figure 2.
The output of a mapping process made by an AMR.
In red, walls and moving obstacles are highlighted thanks to the 3D cameras and sensors of robots.

These devices can be implemented in many different fields. In manufacturing, which is the purpose of this work, they can imply significant improvements, as mentioned in the introductive chapter. AMRs can enable high product mix capabilities and flexibility without necessarily changing the plant layout as they are meant to perform in high-traffic zones and narrow aisles. Redesigning the layout of a productive plant in order to introduce new flexible production line requires heavy investments which could be avoided by the mere introduction of robots allowing the system to acquire more flexibility at a reduced cost and effort. In addition to this, robots can interact with human operators as co-workers in several phases of their use and even assist workers in mounting components in the assembly phase. An example of this is the role they play in the automotive sector, where they operate alongside humans in the assembling of heavy parts.

Autonomous Mobile Robots, as mentioned, can carry heavy loads but travel through narrow aisle and doors at the same time thanks to their small dimensions and their advanced obstacle avoidance system, which gives them the potential of being introduced in many other fields. They can be adapted to many tasks thanks to the possibility to implant many kinds of mechanical add-ons such as cleaning devices or robotic arms. One example is warehousing, where they can be used to help the human worker in order picking operations. This activity is significant because it allows the saving of a large amount of time, since order picking is renowned to be one of the costliest and most time-consuming operations in warehousing.

Relating to more recent emergencies, AMRs can provide a good level of automation in non value added activities such as cleaning and disinfecting rooms, for instance in hospitals. Its technology makes possible to work on every square inch of the floor and to avoid obstacles that, in such a hostile environment as a hospital floor, might occur frequently. They can save time for doctors or nurses by just-in-time deliveries of instruments or medicines and they might also carry out some activities that are now executed by humans but imply risks for human health.

One more application field for Autonomous Mobile Robot might be in safety, thanks to their ability to provide constant stream of video and data to the control center. They can basically be moving cameras that could even work at night with infrared technology. Artificial Intelligence enables robots to detect a suspicious activity and assess whether it might be a possible threat.
Last, they are used in hotels for basic tasks like delivering food and drinks to guests in their rooms or in common areas or take out trash. They can also be used for housekeeping as cleaners of corridors and halls.

Their use is going to face an exponential rise in the next decades since a big variety of application is possible thanks to fundamental features like flexibility or autonomy of decision. Furthermore, the convenience of such devices remains unknown to many. Therefore, AMR are expected to find way more applications than the ones already existing.

3. ANYLOGIC SIMULATION ENVIRONMENT

AnyLogic is a simulation software for business that allows companies to gain deeper insights and optimize complex systems and processes. It's mainly used to reproduce and analyze complex and dynamic environments such as queuing problems, real-time variations of warehouses stocks or optimizing production processes. It can also be used to study transportation systems such as railways or even streets interacting with cars or pedestrians. It is particularly user friendly because it doesn't require much programming in terms of code lines, except for some complicated operations or settings. For this reason, it is preferrable to other simulation platforms like Automod. In AnyLogic, a physical representation of the environment is necessary. For instance, in an assembly line, each workstation must be graphically reproduced, and so do conveyors, products and operators, that are considered resources by the software. Each resource must have the appropriate features and must be placed in the right spot of the grid in order to reproduce the real environment as precisely as possible. In this work, it is extremely important to consider the same distance traveled by AMRs in each different simulation in order to make comparisons between different scenarios possible. Therefore, it is required that the graphical representation of the working environment is exactly coincident with reality.

Along with the graphical reproduction of the simulation environment, the flowchart needs to be built. The flowchart is essential in AnyLogic because it stands for the logical order that events follow. As it's possible to understand from Figure 3, it consists of a sequence of blocks and connections between them. Each block represents an event and it is possible to click on it in order to set all its parameters to consequently characterize the whole process according to the situation that must be reproduced.

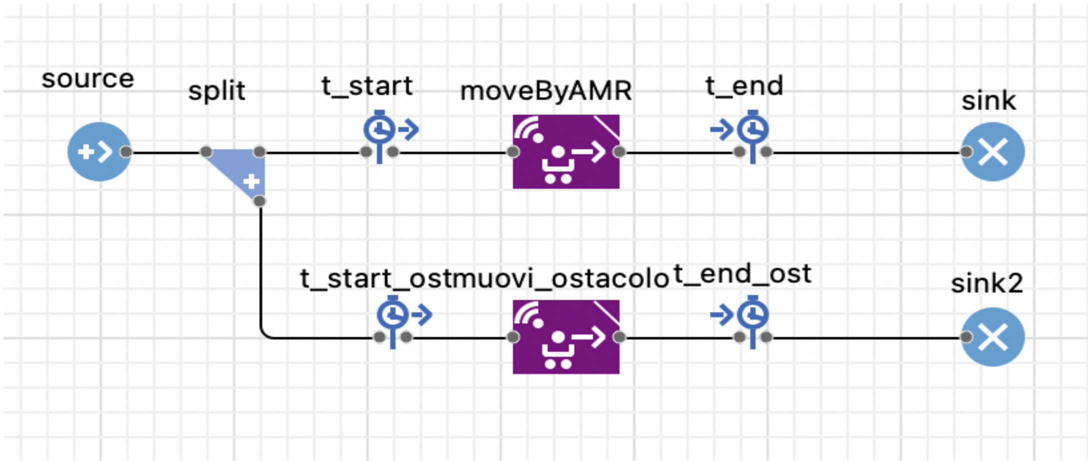


Figure 3. An example of an AnyLogic flowchart.

The flowchart shown in Figure 3 belongs to one of the models created in the making of this work and will be discussed in the next chapters. It contains the most recurring blocks that are significant for the creation of the simulation models.
A list of the main blocks used to create the virtual environment follows.

- Source Block: it consists of a block that triggers the start of the mission. It generates the loading units at a rate that can be set in the setting window according to which is the time interval needed by an AMR to complete a handling mission and pick the following unit.
- Split Block: useful as a logic connector as it allows to synchronize two actions that must be brought on in parallel.
- Sink Block: this block triggers the end of the mission. It is usually not set as its functions are limited but it can often be used as a counter of the units that leave the system. For instance, in this case it counts the output in terms of amount of load units that leave reach the destination point through an AMR transportation.
- MoveByTransporter Block: this is the key to the flowchart because it gives the AMR the instruction for the loading of the item, the pick-up location, and the destination point. In the scenarios that were reproduced in this work, it is significant because it also contains some lines of code that characterize the variables created to measure the mission time. In all the situations where there is a dynamic obstacle, as shown in Figure 3, it is necessary to introduce another MoveByTransporter block in order to create a handling task for such moving obstacle, which might be another AMR, a forklift or an operator.
- TimeMeasureStart and TimeMeasureEnd: these two blocks are placed respectively before and after a block or a series of consecutives block of which

it’s required to know the duration. In this case are placed before and after the MoveByTransporter block in order to measure the duration of the handling mission carried out by the robot.

All the settings of the robot, such as speed, acceleration, size and behavior in proximity of an obstacle are defined in the setting window of another block called TransporterFleet which doesn’t need to be placed in the flowchart area but it’s fundamental to reproduce how the robots interact with the environment into the simulation software.

Even though time measuring blocks were introduced in the flowchart, in this work time will be measured by using two variables based on the occurring of some events in the simulation. The two just mentioned variables were managed through a couple of code lines and are meant to measure time of the travel from starting point to destination point with the vehicle unladen. Therefore, the variables are set as it follows. The first one, called StartWaitingTime, is set to the simulation current time at the exact moment of the unloading of the item at the arrival location. The other variable, called Waiting, is set when the AMR arrives to destination and loads the new unit that needs to be transported and its value is equal to the difference between the current time of the simulation and the value of previously set variable StartWaitingTime. Therefore, Waiting is going to report the time of the return travel of the unladen robot.
This way of setting variables is made possible by an assumption made while setting the parameters of this work: both loading and unloading times are considered as instantaneous activities in order to avoid an influence of loading/unloading times on the travel time.

In this work, as mentioned, the goal is to define a function that associates a certain amount of mission time loss to each circumstance characterizing the work environment. Therefore, for each scenario, it is necessary to measure the mission time of mobile robots and compare it with a base scenario that will be used as standard. The base case environment is shown in Figure 4 and it consists of a straight path from pick-up location to destination point. The length of the path is set to 100 meters and the width of the aisle to 3 or 3,5 meters. All the other scenarios will have the same path length and corridor width in order to allow comparisons.

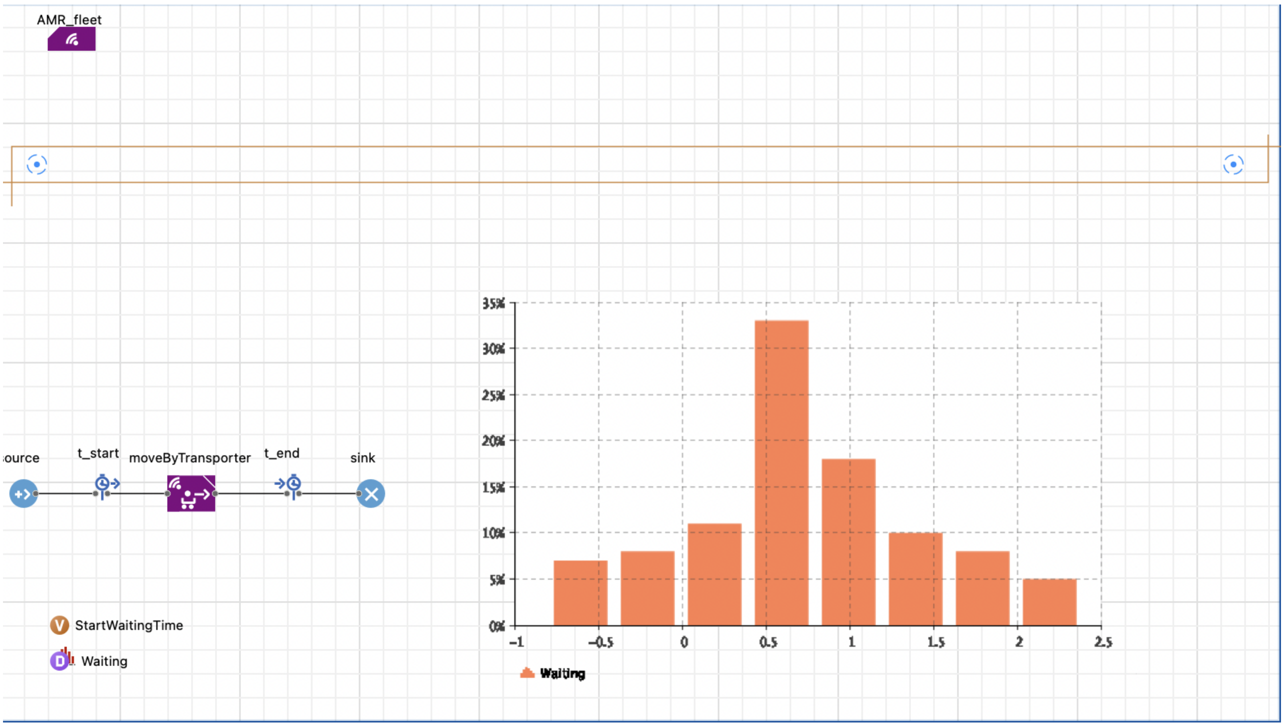


Figure 4. The base case scenario. The 100-meter corridor, on top, is the path traveled by the AMR. On bottom, the flowchart and the graphical representation of the Waiting variable are shown.

4. FACTORS AFFECTING THE MISSION TIME OF AN AMR

The mission time of an Autonomous Mobile Robot, as mentioned in the first paragraphs, is monitored because it has a huge impact on productivity and cost effectiveness. More specifically, it is considered in the fleet dimensioning process and therefore its variations can affect the number of vehicles necessary to implement a successful material handling system. Indeed, the number of robots necessary in a handling fleet is calculated at the condition of knowing the throughput of each AMR, which is described by the following formula,

$$q_{amr} = \frac{3600}{T_c} \cdot C_v \tag{1}$$

where C_v stands for the capacity of the vehicle and T_c for the cycle time of the handling mission. This gives an interesting insight on how important the cycle time T_c is in calculating the throughput of each device and, consequently, in dimensioning the fleet. The cycle time of each robot is calculated as it follows:

$$T_c == \frac{L}{v} \times 2\frac{a}{v} \times 2t_{L/_U} \tag{2}$$

Therefore, the cycle time can be influenced by the speed and acceleration of the vehicles, the loading/unloading time and the length of the path. Since, as mentioned, loading and unloading are not considered in this work and speed and acceleration assume fixed values according to the ones of the robots of NTNU Logistics 4.0 laboratory, the length of the path is the concern of this work. In detail, what affects the above-reported T_c is the equivalent length caused by mission time losses. It is the aim of this chapter to answer to the first key question of this work by detecting and analyzing what factors contribute to a variation of the mission time and therefore a reduction of each device throughput.

A literature review is necessary to find out which are believed to be the most affecting situations in terms of mission time loss by previous research on Autonomous Mobile Robots. From this review it would be predictable to extract factors that influence the distance traveled by robots, e.g. obstacles and their characteristics such as the obstacle width which implies a reduction of the free portion of aisle available for the robot to pass or the speed and direction of the obstacle. Furthermore, there might as well be features of the path itself determining a delay in the mission time.

To perform the literature review about this matter, early research was made on Scopus with keywords like “AMR”, “Autonomous Mobile Robots”, “intralogistics” or “Logistics 4.0”. Nevertheless, the most significant results were collected thanks to further research defined by adding to the previously mentioned keywords more specific ones like “collision avoidance”, “decentralized control” and so on.

The first output of this research is, as predictable, AMR speed. This factor obviously affects the mission time of the robot not by increasing the equivalent distance traveled by the AMR but increasing the value of the denominator of the first term of the cycle time (T_c) formula proposed in the last paragraphs. Although this factor is key to the cycle time of robots, studying how it affects the mission time has little relevance because its value can easily be changed in the AMR settings and it is not related to particular circumstances of the working environment. Furthermore, analyzing it would reveal it as the most affecting parameter and would deviate the focus on those factor that are more relevant to this study. Therefore, for simplicity, it will be kept constant in each simulation in this work at the value found in the laboratory: 0,8 m/s.

One of the most distinctive features of Autonomous Mobile Robots is their obstacle avoidance system. Draganjac explains in his work that these devices move along a wide set of motion primitives, according to its steering limitations. Whenever the vehicle meets an obstacle and senses it, the motion primitive chosen to for the AMR to travel changes and widens as much as needed to get past the obstacle. Consequently, the width of the robot trajectory is proportional to the width of the obstacle, which results to be the second factor affecting the mission time of each vehicle of the fleet. The width of the obstacle cannot be considered alone. In fact, it must be connected to another aspect: how distant the wall is from the obstacle. If the obstacle-wall distance is significantly bigger than the vehicle size then the position of the wall itself can be negligible but, as shown in the following chapters, it has a huge impact if the pass is just a little wider than the robot size. Therefore, the factor affecting the mission time is not the width of the obstacle, it is instead the width of the pass between wall and obstacle.

Draganjac adds the idea of private zones, which allow each robot to privatize every cell of the environment needed to pass and communicate it to other vehicles of the same fleet. If another device is planned to transit in the same cells, a dispatching algorithm will give instructions on priorities to solve the conflict. This makes it possible to understand how necessary it is to consider the possibility of encountering moving obstacles. In case such obstacles are not other vehicles of the same fleet but, as frequently happens, operators or other entities, the robot will face a change of direction which is going to be more unexpected than a robot or a not moving obstacle, since the reaction time is shorter. Therefore, the trajectory computed to avoid it is going to be larger proportionally to the speed of the obstacle. This just mentioned factor is to be taken into consideration.

It is important to know some aspects that distinguish a real AMR by the ideal one that's implemented in AnyLogic simulator. As Liaquat states in his work, the differences between the two concern mainly their behavior in situations where dynamic obstacles meet the AMR, e.g. in a situation when an obstacle is approaching from behind or there are many moving obstacles in sequence that must be overtaken. This aspect is going to be discussed furtherly as a significant limitation of this work and implies future developments like simulating these scenarios in a real environment instead of a simulated one.

One of the situations just mentioned might be considered as a separated factor that affects the mission time of Autonomous Mobile Robots: the frequency of multiple moving obstacles in sequence. When a moving obstacle approaches the AMR, it is overtaken by it and, afterwards, the robot needs to go back to its pre-routed path. If another moving obstacle approaches, the robot must repeat the same operations, which doesn't create any problem if the overtaking procedure occurs a long time after the previous overtaking but might be a problem if the two of them are not much distant in time from one another. This means that, if the frequency with which moving obstacles appear in the AMR obstacle-detecting range increases, then the trajectory to avoid such obstacles widens because the robot doesn't have time to get back to its predetermined path but must deal with another obstacle already. In terms of mission time loss, this factor certainly has an impact and, therefore, must be taken into consideration.

The same approach can be used with static obstacles. Indeed, if many obstacles are placed randomly in a shop floor environment, what can happen is that many of them are placed one after another in a rapid sequence and therefore the trajectory of each device is sharper and, consequently, longer. The frequency of static obstacles will be reported meticulously in this work and, thanks to its reproducibility on the simulation software, it will be modeled into a mission time variation function.

As the last factor affecting the mission time is introduced, it is important to remind that the mission time is affected by those situations that determine an increase or a decrease of the path length, as it was intuitively clear from the T_c formula.
A situational factor that has an impact is the curvature radius of a turn in the path. It can be defined as *the absolute value of the reciprocal of the curvature of a curve at a given point,* and it basically describes how smooth a curve is. The bigger the curvature radius, the smoother the curve. This implies that, taking its value as close to zero as possible, a sharp angle would result.
As it is clear from Figure 5, which is an example on how it impacts on liquids in a piping system, the path is significantly shorter if the curvature radius is longer.

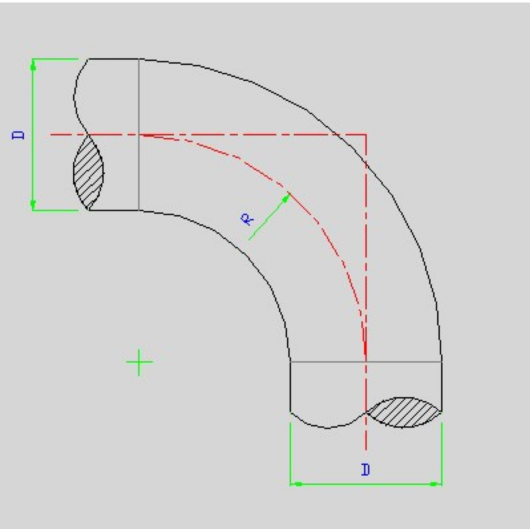


Figure 5. The curvature radius of a pipe. The paths, in red, are shown and compared.

In manufacturing, a circumstance like this is unlikely to occur, since in a production plant there are usually aisles forming sharp angles. However, a similar analysis can be brought on by considering the angle degrees which, changing, can imply a relevant deviation of the robot path. Therefore, since this scenario is closer to a real production layout, angle degrees are considered as the last factor affecting the mission time of an AMR.

To summarize, the factors that were detected as most relevant are:

1. AMR speed.
2. Width of the pass.
3. Obstacle speed.
4. Frequency of dynamic obstacles.
5. Frequency of static obstacle.
6. Angle of curvature.

In the following chapter these factors are singularly analyzed in order to find out if each of them can be modeled as a parameter and used to extract a function that associates to each value of the parameter a value of the mission time variation.

5. PARAMETERS AND SIMULATIONS

In this chapter some considerations are made regarding the key factors that were just discussed in chapter 4 and whether they can be modeled as parameters to create a function that associates their value to a mission time variation.
In order to be consider a parameter, each of the factors detected must verify some conditions:

- They must be an exact or highly close representation of reality. This aspect implies that the times obtained through the simulation software must be correspondent with reality and that AMRs must behave in a realistic way, e.g. in the collision avoidance actions or in determining the minimum distance from an obstacle.
- Variations of their values must cause significant variations of the mission time of Autonomous Mobile Robots.
- They must be relevant to this work.

Each factor is now shortly discussed and the reasons of the choices are highlighted.

All the parameters that are introduced in this chapter will undergo further examinations that will consist of simulations, computing a function that associates each parameter value variations to a mission time variation and a short conclusive discussion. What is to be done before these steps is conferring each parameter a range of values that will be taken as base values of the simulation. In the making of this analysis, the obstacle speed is considered even if it is not a parameter and its range of values is limited to the values that make it possible to compare it to reality and compute a function characterized by a realistic trend.

In the following table are reported the range of values for each parameter and factor.

	PARAMETERS	Unit of measure	Base value	Minimum value	Maximum value
1	Width of the pass	Meters	1,25	0,95	1,5
2	Frequency of static obstacles	Meters/obstacle	15	2,5	50
3	Obstacle speed	Meters/second	-1,5	0,5	2

4	Curvature angle	Degrees (°)	90	45	135

Table 1. The three parameters are reported, adding the obstacle speed because of its characteristic of being computable into a function within the strict limit of values that allow a realistic representation of reality.

5.1 INTRODUCTION TO RESULTS INTERPRETATION

As anticipated, this chapter also focuses on the output dataset for each parameter. All the measurements are collected in an Excel file, in which the most relevant simulation results are reordered and the other values are rejected. The following step is computing a function relating each value of each parameter to its result, which is its mission time variation compared to the base scenario.
This process can be carried out by using MatLab, a data analysis application used to develop algorithms or create models and functions. Out of the many tools developed by this software, the one that is significant for the aim of this part of the work is the Curve Fitting Tool. Such tool allows to fit curves and surfaces to externally provided data. Therefore, it is possible to conduct regressions analysis using the library of linear and non-linear models provided. Although they were not used in the making of this work, the Curve Fitting Tool also supports nonparametric modeling techniques, such as splines, smoothing and interpolation.
Once the data provided by the simulation are inserted in the software, a function is computed and is shown in a Cartesian graph. In it, the X-axis the values of the parameter are reported while in the Y-axis the mission time variations are shown.
The model firstly computes a function, whose type can be changed by the user to find the one that has the best fit. Computing the parameters described in Chapter 6, several models resulted to be well fitting the input data, determining the need for choosing one for each parameter.

Having said that, the question that raises is the following: how is it possible to state which function better fits the input data? What is the evidence that justify this choice? MATLAB answer these questions by providing the user with useful indicators that measure the error of the just created model and therefore describe the goodness of its fit. Such indicators are SSE (), R^2, R^2_{adj} and RMSE and are defined as it follows.

- SSE stands for sum of squares error and it measures the goodness of fit of a regression model by calculating the sum of the square of each deviation of the actual value from the fitted value. Its formula is the following.

$$SSE = \sum_{i=1}^{N} e^2$$

- R^2 is a goodness-of-fit measure for linear regression models. This statistic indicates the percentage of the variance in the dependent variable that the independent variables explain collectively. R-squared measures the strength of the relationship between the model and the dependent variable on a convenient 0 to 100 percent scale. This measure considers the smallest sum of squared residuals that is possible for the dataset that the user has inserted as model input. A residual is the difference between the observed value and the fitted value. Since R^2 is a numeric result, it cannot be considered valid unless residual plots are evaluated. For instance, the result can be acceptable according to R^2 even if some of the residuals were systematically too far from the fitted values. When a scenario like this happens, this measure must not be considered effective because the model is biased. Therefore, it is appropriate to carry out a residual plot analysis before validating the R^2 result.

  Having given such assumptions, it is possible to continue with the analysis of R2 by computing the following formula.

$$\boldsymbol{R^2 = 1 - \frac{variance\ unexplained\ by\ the\ model}{total\ variance}}$$

  As the formula might suggest, this indicator expresses a better fit of the model if its value is closer to 100%. In this work and in general, a curve is considered well fit to the input dataset if R2 is greater than or equal to 95%.

- R^2_{adj} is an indicator which improves what R^2 measured. It is useful in those cases in which more variables are considered in a model. It keeps the number of variables (K) and the number of points in the data sample (N).
  The formula for adjusted R-squared is the following.

$$R_{adj}^2 = 1 - \left[\frac{(1-R^2)(n-1)}{n-k-1}\right]$$

This indicator corrects a problem that often happens by using R^2 , which is its tendency to increase with more terms being added to the model. This is completely misleading.

- RMSE, or root-mean-square error, indicates the goodness of the fit if its value is low. It is basically the quadratic mean of all the deviations (errors) between the observed and the fitted values.

6. AMR SPEED

6.1 PARAMETER PRESENTATION

The speed of robot was mentioned in the previous paragraphs as a factor that influences the mission time in a direct and clear way. It is reported as the denominator of the first factor of the T_c formula (2) and therefore a variation of its value directly determines a variation of the mission time of an AMR.

From simulating in AnyLogic and from the experiences in the NTNU Logistics 4.0 laboratory, it was possible to verify that the speed of these devices decreases every time they meet an obstacle or they approach a turning point. Observing and measuring their paths in the lab, the minimum distance to an obstacle was believed to be 0,5 meters. Measuring times and comparing them to the ones detected from AnyLogic, it was possible to extract the values of acceleration and deceleration that were set to 0,25 m/s. Furthermore, the user interface of the AMRs used in the lab automatically set its speed to 0,8 m/s.

As mentioned, this factor verifies the first two conditions but not the third one and, therefore, will not be considered a parameter in the making of this work mainly because it is pointless to change its value knowing its real speed in the working environment. Moreover, its impact on the mission time is incredibly higher than any other parameter of this work and consequently the results would have been excessively altered by this specific factor.

Last, this work highlights the behavior of mobile robots considering particular reality-like circumstances of the working environment rather than focusing on the settings of the devices.

7. WIDTH OF THE PASS

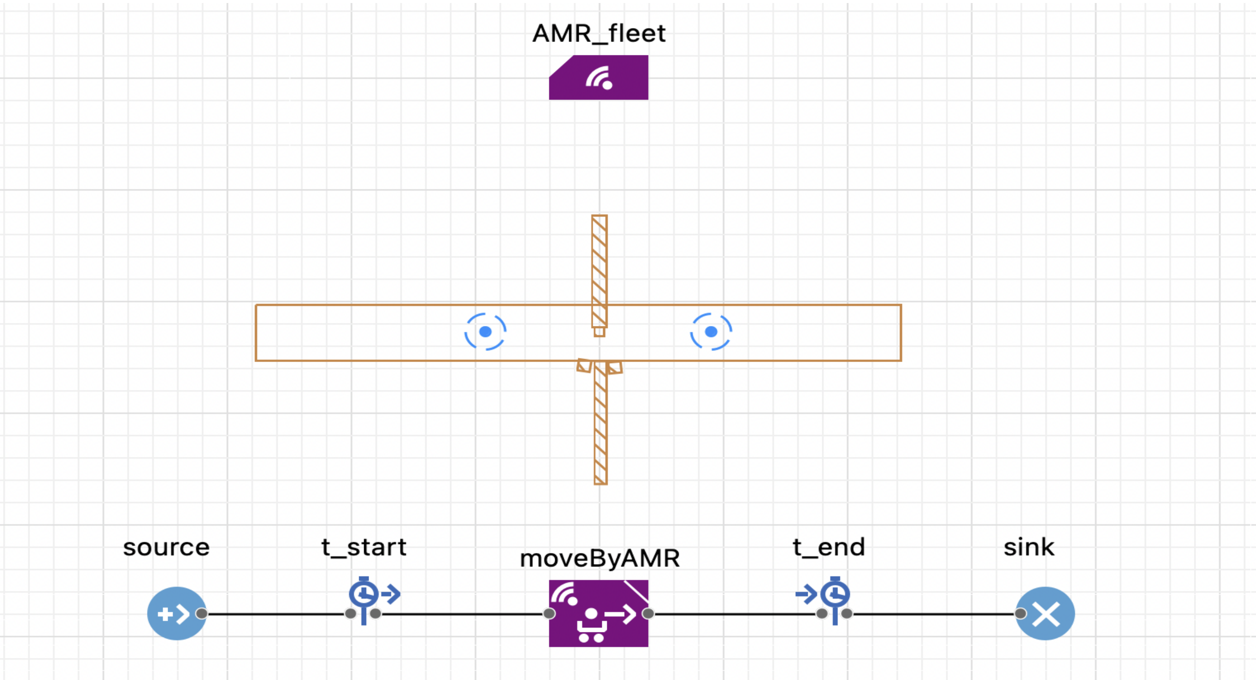
7.1 PARAMETER PRESENTATION

This factor is the result of the combination between the position of the obstacle and the wall, and the differences between these values generates the width of the pass available to let the robot get through. In this case, it was necessary to reproduce each scenario in the laboratory because of significant differences from the behavior of the AMR while approaching and overtaking the narrowing in AnyLogic. Indeed, in the simulation software the robot was able to get through a 58-centimeter large narrowing, which is exactly the robot size in width. In reality, as stated in the manual of instruction of MiR200, the device needs at least 90 centimeters to get through, which is exactly what resulted from the experiments brought on with PhD. Mirco Peron.

For this reason, the first condition is satisfied, as well as the second and the third. Therefore, this factor is to be considered a parameter.

During the experiments, more than one scenario was reproduced and, in the end, the most relevant one was determined to be the one characterized by a square obstacle (40x40), approximately the width of an operator, placed in the middle of the robot path. The width of the pass was varied during the simulations to allow to monitor the mission time variations under each circumstance.

7.2 SIMULATION WITH ISOLATED PARAMETER



fi.

The simulation environment for the width of the pass parameter.

This parameter was the only one which was entirely measured in the NTNU Logistics 4.0 laboratory, thanks to the great patience and availability of PhD. Mirco Peron and the permission of Professor Fabio Sgarbossa. In the lab, a box which was approximately the size of a person was placed in the middle of the aisle and the vehicle was forced to avoid it passing on the right side of the aisle. The pass was reduced more and more each time until it was 0,9 meters wide. In that scenario, the robot could only pass few times out of more than 20 tries, and therefore the measurement was not considered a valid result. Moreover, the experiments in the lab showed that there was no difference in terms of mission time if the width of the pass was equal to 1,5 meters or bigger. These are the reasons why the values were decided to range between 0,95 to 1,5 meters, which resulted in a significant impact on the mission time, as it is shown in the following chapter.

Although most of the job was made in the laboratory, each value of the previously set range was reproduced in the simulation software.
As shown in the picture, the simulation environment for this parameter is set as follows. The main aisle is 100-meter long and 3,5 meters long. In the picture above it is set to 20 meters to make it fit in the figure and to show both the starting and ending points. In the middle of the path there is a static obstacle in the upper part of the aisle which remains still. The moving part, which increases and decreases the width of the pass, is the lower part, consisting of another obstacle whose function is just restricting or enlarging the narrowing.

7.3 SIMULATION RESULTS

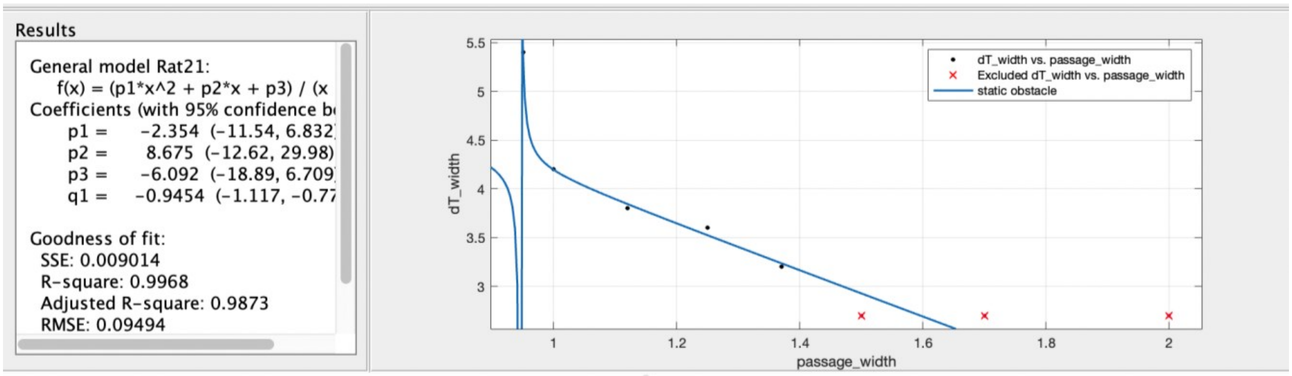


Figure 7. The mission time variation function for the factor “width of the pass”. This is the model that is valid for passage width values that are less than or equal to 1,5 m.

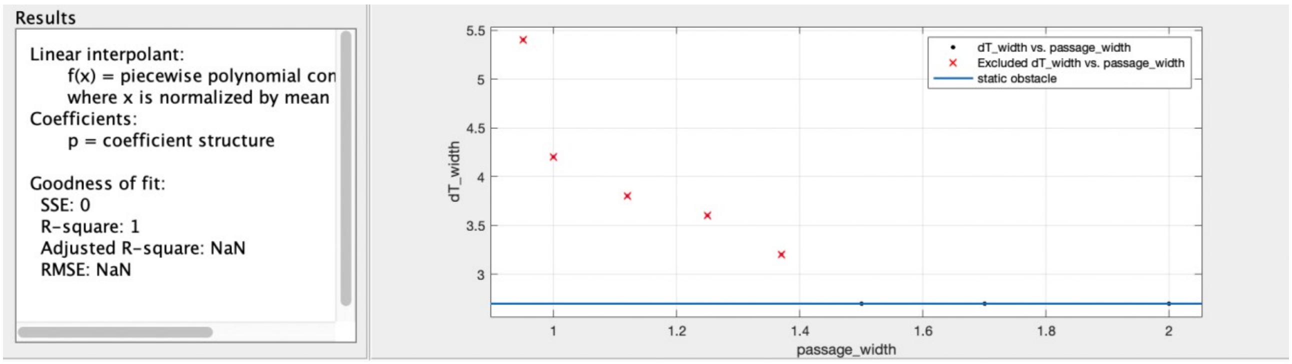


Figure 8. For passage_width values greater than 1,5 meters, the function is a straight line whose value on the Y-axis remains constant.

As it is possible to see in Figure 7, the function computed by the Curve Fitting tool perfectly describes the model in that range of values going from 0,95 to 1,5 meters.
This first graph was limited to this range of values. Indeed, the graph shows a function that keeps decreasing as the input parameter increases even for bigger values. This goes against the deductions made while considering the results of the lab experiments, which revealed that the mission time variation does not increase when the passage width is greater than 1,5 meters.
Therefore, in this case another graph is needed to show the trend of the function for passages wider than 1,5 meters, which is basically a constant value. This trend is showed in Figure 8.
The following formula quantitatively describes the model for values on the X-axis lower than or equal to 1,5.

$$\Delta T = \frac{p_1 x^2 + p_2 x + p_3}{x + q_1}$$

Where:

- x = passage width value.
- p1 = -2,354
- p2 = 8,675
- p3 = -6,092
- q1 = -0,9454

8. FREQUENCY OF STATIC OBSTACLES

8.1 PARAMETER PRESENTATION

The frequency of static obstacles is a relevant factor because it reproduces a scenario that can be easily found in a real working environment. For instance, in a long aisle with multiple working stations there might be obstacles of any kind, from pallets to operators, from stationary vehicles to semi-finished products. As stated in the previous chapter, their presence could be considered as a single obstacle scenario if the distance between obstacles is high enough. What happens if such distance is shorter is that AMRs are forced to travel more sharp paths, enlarging their trajectory in order to avoid consecutive obstacles that imply consecutive changes of directions. This factor is therefore relevant, can be modeled through the simulation software reproducing reality with great coherence and its variation cause significant variations in terms of mission time. For all the reasons which were just mentioned, this is going to be another parameter of this work.

8.2 SIMULATION WITH ISOLATED PARAMETER

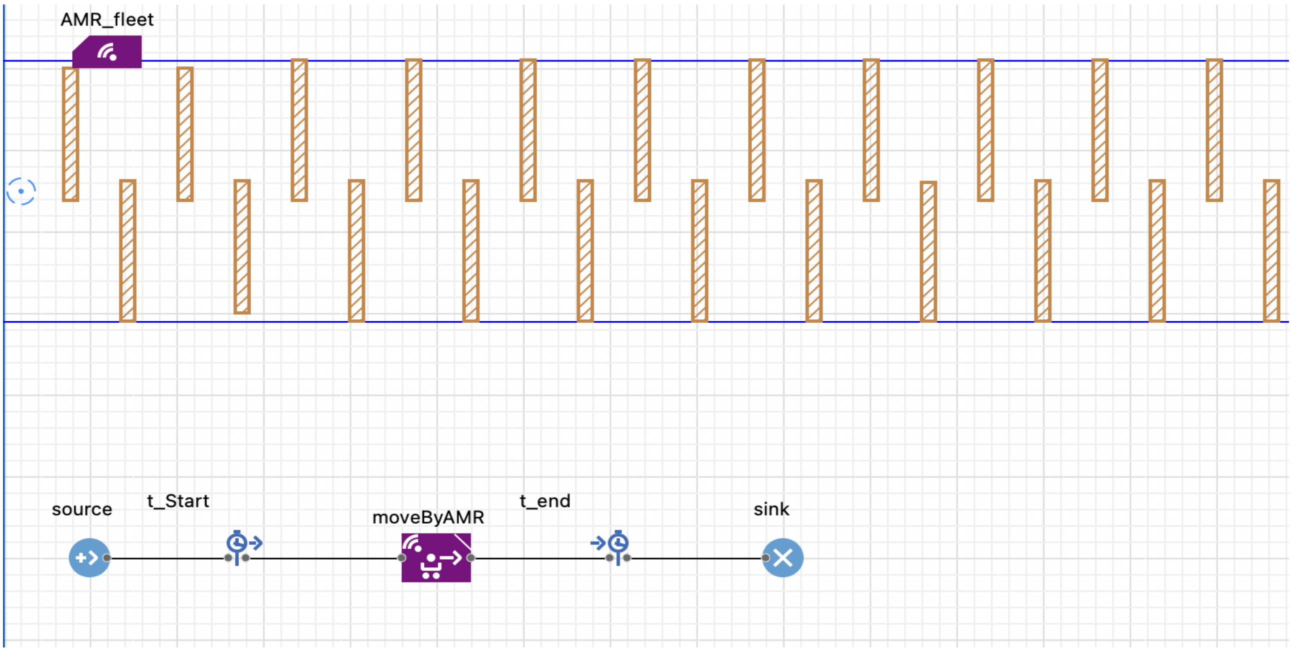


Figure 9. Simulation environment for the frequency of static obstacle.

This parameter, unlike the previous one, is entirely simulated in AnyLogic because it is needed a longer aisle to variate the frequency of obstacles. The minimum value is set to 2,5 m/obstacle, which means that the Autonomous Mobile Robot finds an obstacle every 2,5 meters. This situation is rarely found in a working environment, but it allows to enlarge the dataset needed to compute the mission time variation

function. The maximum value is 50 m/obstacle, which results in only one obstacle in the path because, as stated in the previous chapters, the aisle is 100 meters long. The experiment was brought on considering both a person size obstacle and a pallet size obstacle. The results, in terms of mission time variation, were quite similar. In an industrial context in which a situation like this can happen, the aisle is often wider than 3,5 meters, otherwise there would not be enough space for the robots to overtake a pallet, facing the device by the long side, and get through the narrowing formed by the wall and the obstacle. Therefore, in the simulations the aisle width is set to a bigger value, as shown in Figure 9. Obstacles were enlarged on one side, one time to the left and one to the right, in order to force robots to travel a "zigzag" trajectory, otherwise they would avoid every obstacle by the same side without going back to the center of the path.
To summarize, starting from one obstacle, another one is added in every simulation, changing the position on the grid of each obstacle to keep constant the distance between them.

8.3 SIMULATION RESULTS

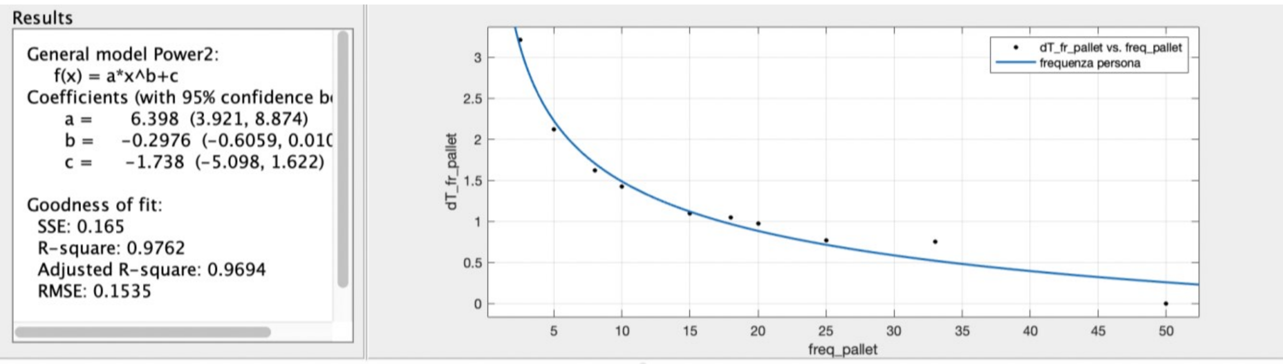


Figure 10. The trend of the function of the parameter "frequency of static obstacle".

The frequency of static obstacle is modeled as Figure 10 shows. The dT does not refer to the overall mission time variation. On the contrary, it refers to the mission time variation caused by each obstacle the robot finds on its path. For instance, as the graph shows, if the AMR must overtake one obstacle every 5 meters, the mission time variation is 2,2 s/obstacle. This value, then, must be multiplied by the total number of obstacles in the path to find the total amount of mission time lost in the whole path. This decreasing trend is explained by the fact that when many obstacles are on the robot way, it must make many more changes of direction. Moreover, each change of direction is sharper and therefore requires more time to be executed.

This experiment was brought on considering both an obstacle the size of a pallet and a person. The results are similar in terms of values and even in terms of the function computed by MATLAB. The only difference that is brought to attention involves the smallest value of the frequency range: 2,5 m/obstacle. In this scenario, pallets determine a 3,2 second mission time variation while obstacles the size of a person cause a 2,16 second variation. However, it is extremely unlikely for a robot to find one obstacle every 2,5 meters in a real-life working environment and therefore it is not fundamental to distinguish the two previously mentioned cases.
The function returned is a negative power relation, described by the following formula.

$$\Delta T = ax^b + c$$

Where:

- a = 6,398
- b = - 0,2976
- c = - 1,738
- x = value of the frequency of the static obstacle.

As it is possible to notice by observing the results reported in Figure 10, the statistics confirm the goodness of the model by highlighting a R^2 indicator that is nearly 98% and a low value of the SSE, which is equal to 0,165.

9. OBSTACLE SPEED

9.1 PARAMETER PRESENTATION

An Autonomous Mobile Robot can avoid both static and dynamic obstacles, approaching both from the back and from the front. This characteristic gives these devices great flexibility but, at the same time, makes them behave differently from a static obstacle situation. Indeed, when an obstacle is moving towards the robot, it is sensed by motion sensors and by cameras implanted in the device but it reduces the reaction time of the AMR and, above all, it forces the robot to enlarge its trajectory because it must let the aisle free for the obstacle to pass in a shorter time. Moreover, as stated previously, in some scenarios the obstacle might come from the same direction as the robot and, if its speed is higher, it must overtake the robot. In this case the robot must stop to let the obstacle get through and only afterwards it can continue its mission. In AnyLogic a dynamic obstacle scenario cannot be reproduced

for one reason: when an obstacle moves towards the AMR with a much higher speed than the robot speed, the software behaves in an unrealistic way, forcing the robot to move out the way to avoid colliding with the obstacle at a speed which is not realistic at all. As result, the AMR mission time is shorter when an obstacle approaches at a high speed (2,5 or 3 m/s, the speed of a random forklift). It could be considered as an obstacle only if the obstacle speed is less than 2 m/s.
Considering these observations that have just emerged, the behavior of the simulation software becomes a highly significant limitation of this work and further research based on laboratory experiments is appropriate.

9.2 SIMULATION WITH ISOLATED PARAMETER

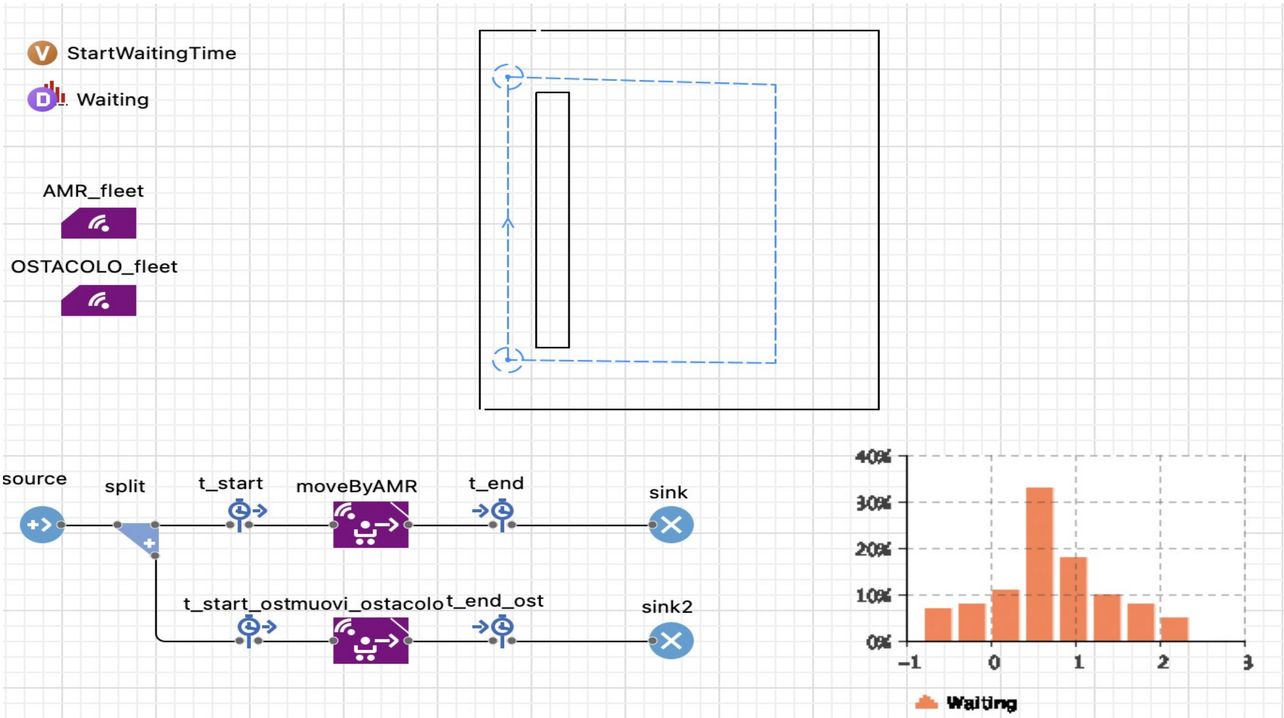


Figure 11. The simulation environment of the obstacle speed factor.

This factor, which was previously discussed as a non-parameter for this work, is one of the most complex to represent because the simulation environment is modeled aiming to guarantee that the robot and the obstacle meet at the same point of the path in each simulation to make sure they meet as many times as the number of missions of the AMR. In figure 11figure it is shown the model and the blue path, which is a trajectory the obstacle must not leave. Indeed, in this case obstacles are transporter entities just like AMRs but are set as path-guided ones. Such path is

modeled in length to achieve what was just mentioned: the coincidence of the robot and the obstacle path in each loop of the simulation.
Furthermore, AnyLogic gives the user the possibility to change the direction of an already existing path, which is what is done to carry out the simulation of the obstacle moving in the same direction as the vehicle. The obstacle can be modeled in width, length and speed in order to reproduce different vehicles interfering with the robot mission: forklifts, operators, AGVs or even other robots.
Just like the previous cases, in the picture is reported a working area in scale to fit the figure.

9.3 SIMULATION RESULTS

As stated in the previous chapter, this factor is not considered a parameter for multiple reasons, one of which is because the AMR responds to an obstacle approaching in a realistic way only for speed values ranging from 0 to 2 m/s. When an obstacle is faster or when it approaches the robot from the back, the device represented on AnyLogic does not match the behavior or real-life AMRs. Therefore, future developments on this are appropriate and further experiments in the laboratory are necessary.
Having said that, the trend of the dT function is reported, for the sake of completeness, considering only positive values from 0 to 2 m/s.

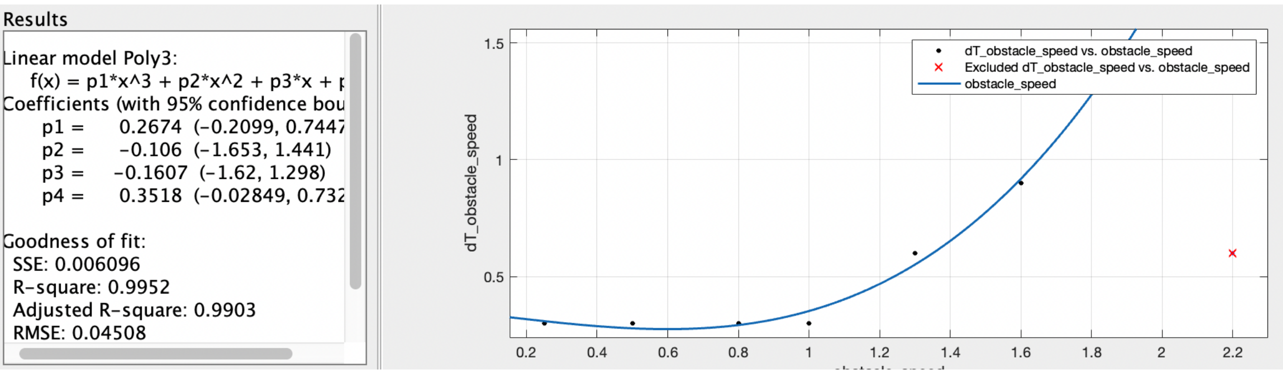


Figure 12. The function trend for the factor “obstacle speed”.

As it is possible to notice from Figure 12figure, the function keeps raising as the obstacle speed increases because the observed value at obstacle speed equal to 2,2 was excluded (red cross on the right part of the graph) because the software simulation reproduced an obstacle avoidance process which was not realistic. Indeed, if that value had been considered, the function would have unpredictively collapsed even though the obstacle speed was increasing. Therefore, it is necessary to limit the obstacle speed values to 2. Future developments are needed to allow this scenario to be modeled and a reliable function computed for obstacle speed values even higher than 2 m/s. Indeed, these values of speed are realistic in a working environment, since forklifts, for instance, can travel at a speed up to 3,5 m/s.

10.FREQUENCY OF DYNAMIC OBSTACLES

10.1 PARAMETER PRESENTATION

This factor is conceptually similar to the frequency of static obstacles. Indeed, the way the robot behaves is comparable because it avoids the obstacle and then moves back to the center of the predetermined path and avoids the collision with the following obstacle. It is coherent with reality since there are plenty of situations where robots encounter pedestrians in a relevant frequency or even other vehicles like other AMRs or even forklifts. As it is possible to notice thanks to the simulations made with static obstacles in sequence, the waste of mission time these situations imply is relevant and could be modeled as a parameter. In this case, what prevents this factor from being considered a parameter is its low reproducibility on the software AnyLogic. The problem detected during the simulation concerned the path of the robot: while in the static obstacle scenario it was possible to lead the robot to a "zigzag" path to perform many changes of directions, in this scenario it is not possible due to the dynamicity of the system. In other words, the AMR does not go back to the center of the path but stays on one side of the aisle in order to avoid obstacle in a smarter but not realistic way, as shown in Figure 13.

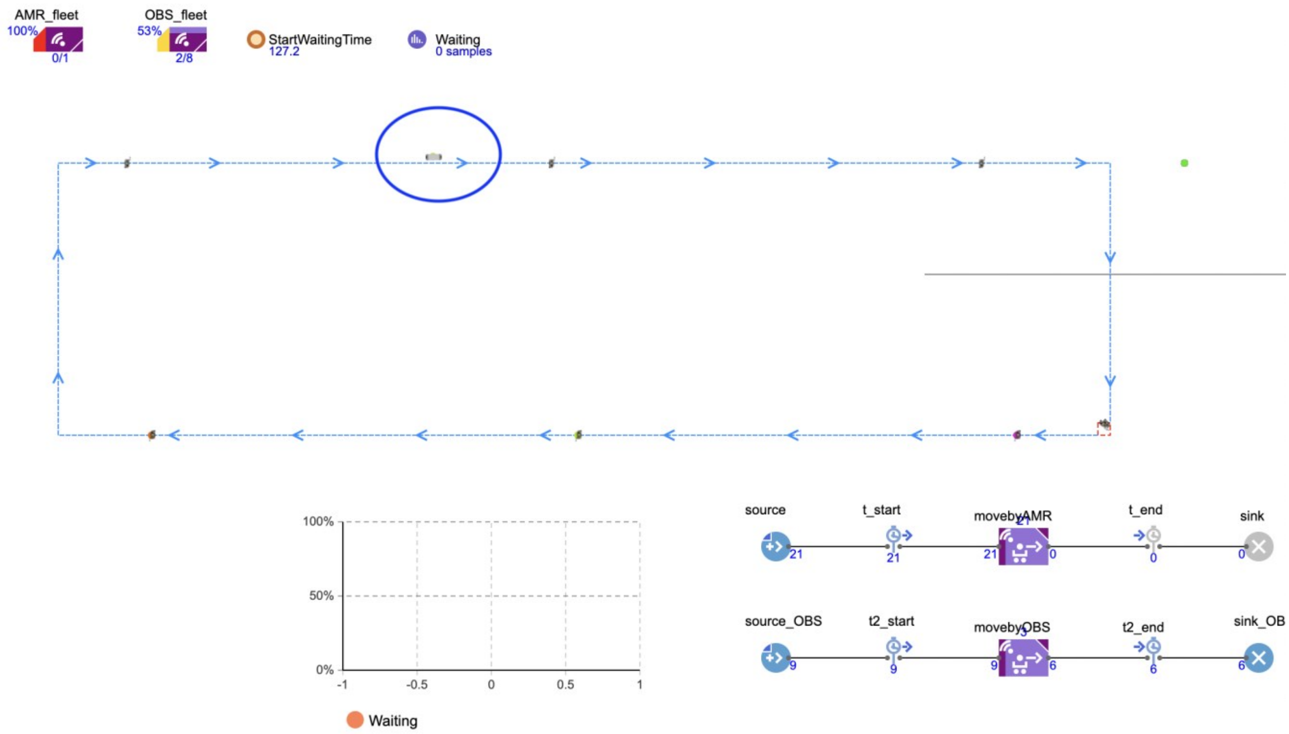


Figure 13. As shown, the AMR (circled in blue) does not come back to its preset path (line in light blue) after avoiding the first obstacle (moving operator on the blue path) but it keeps the right to make it easier to avoid the following obstacle. Because of this robot characteristic, it is not possible to reproduce in AnyLogic a realistic simulation environment.

11. CURVATURE ANGLE

11.1 PARAMETER PRESENTATION

The curvature angle is the manufacturing translation of the curvature radius in piping systems that was detected during the literature review. This change was made mainly to reflect the curvature radius to a more realistic environment to be reproduced. It is significant to this work because it implies serious waste of mission time if its value variates. Operating in the laboratory was a critical advantage because it allowed to understand how close robots got to the walls during a 90 degree turn and how much their speed decreased in that part of the path. Thanks to what was observed in NTNU Logistics 4.0 laboratory, all the fleet parameters were set in AnyLogic as well. Since the curvature angle is relevant and can be reproduced with good coherence with reality, it is considered a parameter, which is expected to imply higher variations of robot mission time when the angle is smaller and lower variations when the angle between the aisle is tending to 180°.

11.2 SIMULATION WITH ISOLATED PARAMETER

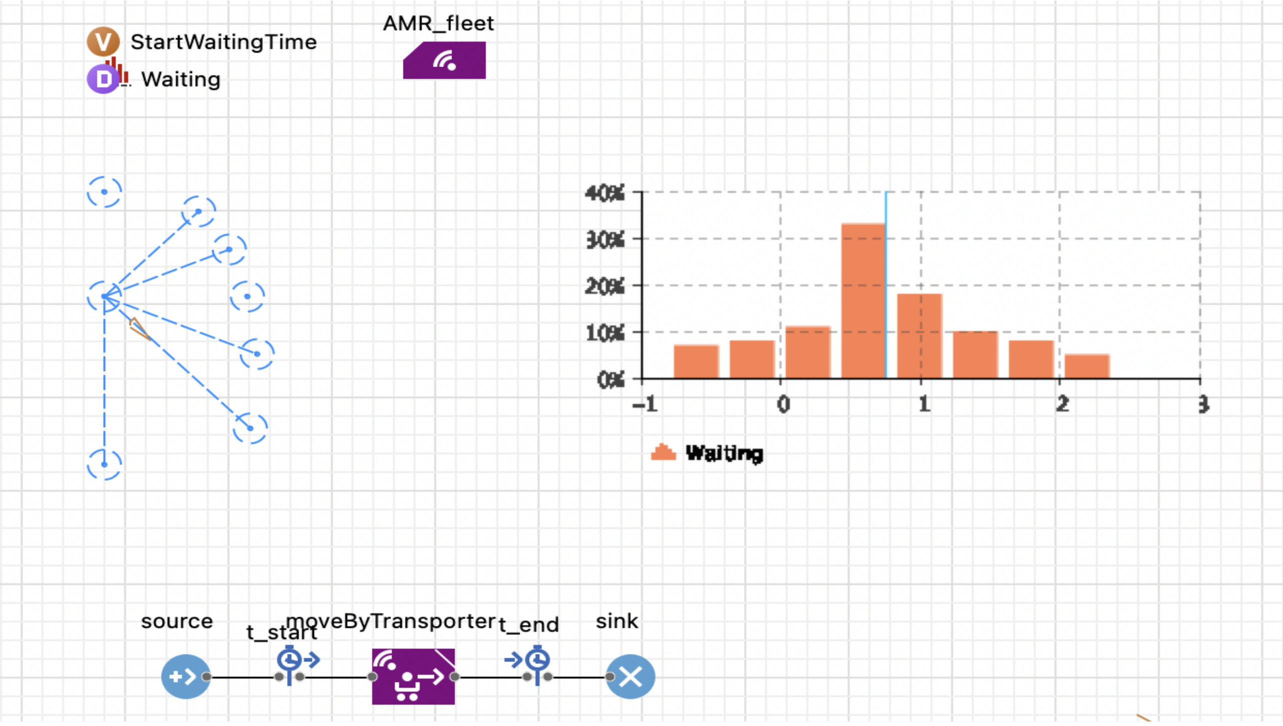


Figure 14. The simulation environment of the factor curvature angle.

In the picture are shown, in blue, the main vertical path, which is common to all the 5 values of the range, and the 5 different paths forming 5 different angles: 45°, 67,5°, 90°, 112,5° and 135°. Again, the lengths are scaled to make them fit in the picture.
The most difficult part of this modeling process was to make sure that each path was equal to the others. In the picture it is possible to see that the blue paths are far from being the same length, and that is because AMRs don't move precisely along the pre-determined path but are set as free-space movers. Therefore, when two aisles form a small angle, the AMR will not follow the path in blue, it will cut the turn and pass as close to the wall as possible, which was observed to be 50 centimeters, thanks to its artificial intelligence that computes the fastest trajectory to reach the destination point. Whenever the angle formed by the two aisles is bigger, for instance 135, the robot will act the same way, but it will not be possible for it to save as much space as in the previous cases because its trajectory will be closer to 180°. This is the reason why the smaller angles are characterized by longer paths and further destination point. Such distance is calculated by observing the trajectories of the robots in AnyLogic and making sure that each path results in the same distance traveled by the device. In Figure 14 walls are not shown because their position varies along with the

variation of the scenario: the positioning of walls when the 45-degree angle is considered is different than when the 112,5-degree angle is simulated.

11.3 SIMULATION RESULTS

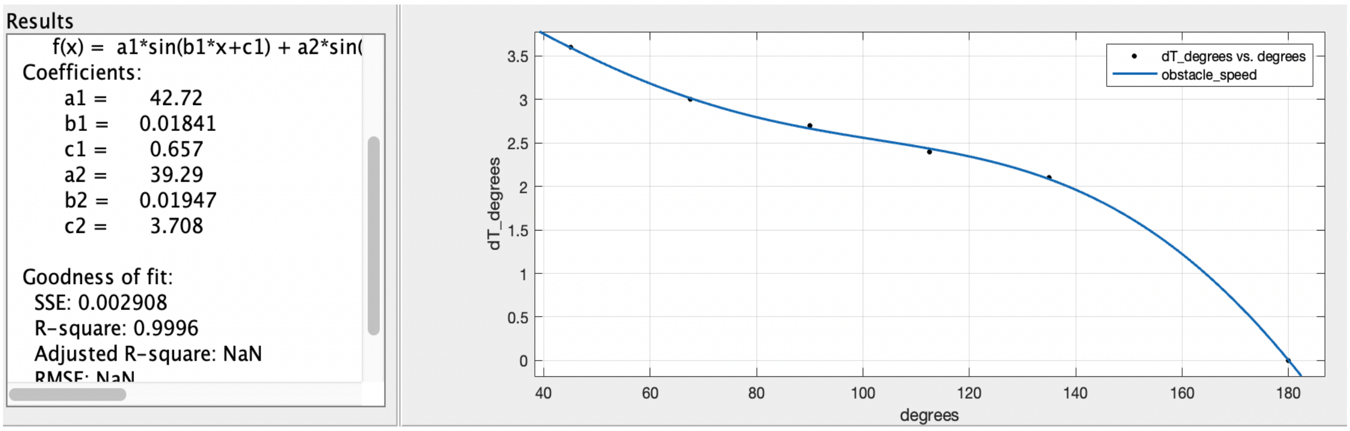


Figure 15. The computed function of the parameter "curvature angle".

The graph shows a trend determine by a sum of sine kind of function. The computed formula describes the input data very accurately, with an SSE of just 0.003, while the R^2 indicator nearly reaches 100%. Obviously, the mission time variation when the curvature angle equals 180° is null. Therefore, the graph should be stopped at that value on the X-axis and the function shall not have negative values.

In this scenario, more than one function described the model properly with good fitting values. In particular, a third-degree polynomial function returned similar R-squared and SSE values, but the sum of sine function was preferred because of its smaller number of terms.

The computed formula is the following.

$$\Delta T = a_1 \sin(b_1 x + c_1) + a_2 \sin(b_2 x + c_2)$$

Where:

- a1 = 42,72.
- b1 = 0,01841.
- c1 = 0,657.
- a2 = 39,29.
- b2 = 0,01947.
- c2 = 3,708.

The results obtained from the simulations are, as expected, more significant when the angle is smaller. Nevertheless, even when a 135-degree angle is involved, the mission time variation is significant and reaches slightly more than 2 seconds. This can be explained by the AMR behavior, since the robot calculates the shortest path to reach the destination point and therefore gets as close as possible to the wall in a turn. By doing this, the robot slightly slows down when it approaches an obstacle or, as in this case, a wall, as observed in NTNU Logistics 4.0 laboratory.

12.COMPARISON OF THE IMPACT OF EACH PARAMETER

The last bit of this chapter consists of a short analysis to check the importance of parameters in terms of how much each of them affects the mission time of an Autonomous Mobile Robot. The activity that must be brought on is a comparison between the mission time loss of each of the three parameters. Indeed, the obstacle speed is not going to be considered because of the limitations that restrain the applicability of the results obtained by simulating with AnyLogic, which are already stated to be a limited range of results.
Moreover, a further distinction is required: the frequency of static obstacles returns a mission time loss value which is comparable to the other parameters if the mission time per obstacle is considered. This is what was considered over the last chapters and in the above-reported computed function.
Otherwise, if the total mission time loss is considered, its value raises exponentially to reach values that are incomparable to the other two parameters.
Both scenarios must be considered because both reproduce realistic situations in a working environment. The comparisons between the impact of each parameter are shown in Figure 16 and Figure 17.

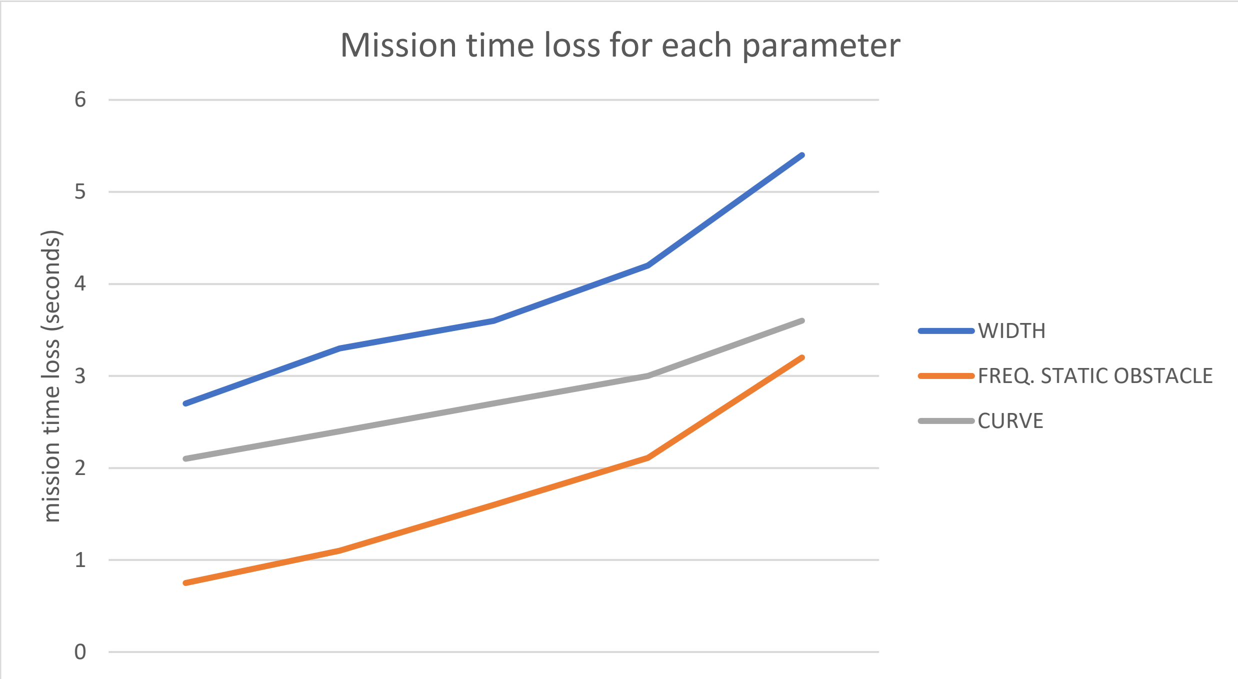


Figure 16. Comparison between the impact of each parameter on the mission time. On the Y-axis is reported the mission time loss in seconds.

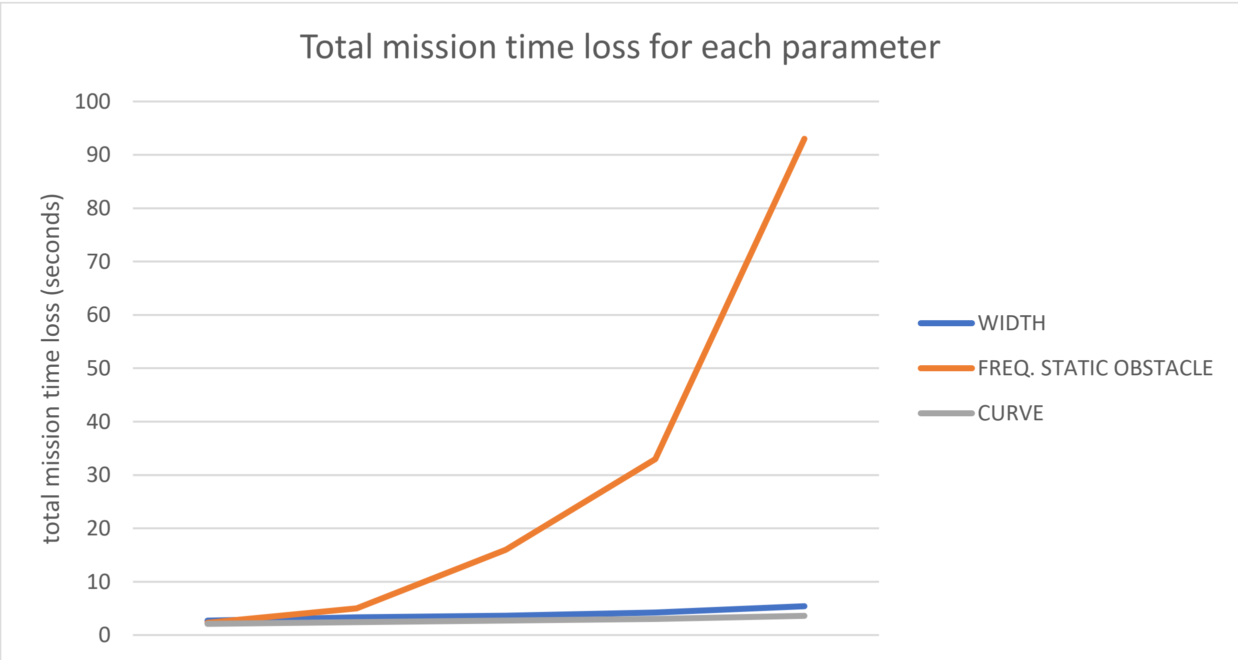


Figure 17. Comparison between the impact of each parameter on the mission time. In this case, for the frequency of static obstacles, the total mission time loss is considered multiplying the time loss per obstacle by the total number of obstacles in the path.

As it is shown in the graph of Figure 17, when considering the total mission time loss, in the worst-case scenario where obstacles are found in the path each 2,5 meters, the frequency of static obstacles rapidly raises to reach 91,2% of the total lost time.
On the other hand, when considering one event at a time and the maximum value that each parameter can get, the percentage of the impacts are more balanced.

The width of the obstacle impacts on the 44,3% of the total, while the frequency of static obstacles and the curvature angle respectively impact for 26,2% and 29,5% of the total.

13.SIMULATION ENVIRONMENT WITH COMBINED PARAMETERS.

The analysis brought on in the previous chapter has returned some important results that have a central role in this work and allow to model many real-life situations and take into account time losses that enlarge the equivalent distance traveled by Autonomous Mobile Robots.
Having said that, it is now fundamental to acknowledge that most of the scenarios of real working environments have not been considered yet. For instance, what happens when an AMR finds multiple obstacles and a curve in the same path? Basically, the situations left to analyze are resulting from the combination of the parameters that were discussed about in the previous chapters. Therefore, in this part of the work, the focus is on creating simulation models in which each parameter is not taken singularly anymore. On the contrary, it is combined with the other two in order to create and simulate more complex models that are more likely to characterize real-life scenarios. According to the assumptions just specified, it is necessary to select some values in each parameter value range and create every possible combination of those to create a three-value model that is going to represent a realistic scenario. In the following table, Table 2, the most significant values are chosen based on how relevant they are in terms of possibility to be found in a working plant and of how much they impact the mission time of an AMR.

WIDTH OF THE PASS (m)	FREQUENCY OF STATIC OBSTACLES (m/obs)	CURVATURE ANGLE (°degrees)
1	25	45
1,25	15	90
1,5	5	135

Table 2. Significant values considered for each parameter. Such values are to be combined to form three-value models.

Concerning the width of the pass, three values out of the four that were previously considered are chosen to be simulated in this chapter. As of the curvature angle, the most realistic values were chosen, as well as for the frequency of static obstacles. Regarding this parameter, the 2,5 m/obstacle value was left out of this analysis because of two reasons: its unrealism and the enormous impact it would have on the mission time. Indeed, if such value was considered, it would influence the analysis and consequently reduce the impact of the other two parameters on the overall simulation time. Its impact is much more significant than the other two factors, as it is possible to notice from Figure 17 of Chapter 7.4.

Since three values per parameter were picked, the overall number of scenarios representable is 27. Indeed, the total number of combinations is returned by a power whose base is the number of factors considered and the index of power equals the number of values per factor. Therefore, in this case, it is 3^3.

Each combination of values determines a situation that will be implemented in AnyLogic to create a virtual environment. The running of the simulation returning a mission time value allows to compare such value to the mission time of the base case and calculate the time difference dT.

Before proceeding to the creation of the simulation environments, it is important to specify what the term "base case" means. The base case scenario is the referring point of the whole set of simulation and the whole data set resulting from it. Each simulation result must be compared to the base case. Such scenario is supposed to reproduce the simplest realistic situation that could be found in an intralogistics plant and, therefore, each comparison shall result in a robot mission time variation that differs in relation to the simulation values selected. The following table sets the three parameters' values of the base case scenario.

WIDTH OF THE PASS	FREQUENCY OF STATIC OBSTACLES	CURVATURE ANGLE
2 m	50 m/obstacle	180°

Table 3. The base case scenario.

As shown in the table, the above-mentioned base case scenario is basically a straight aisle (180° angle) with a single obstacle in the middle of it (50 m/obstacle). The width existing between the wall and the obstacle is set to 2 meters. This scenario is the first one to be reproduced. Starting from this, the 29 simulations are going to be carried out introducing a new curve or new obstacle and dislocating existing ones. Moreover, walls need to be moved as well, making the width of the pass smaller according to the values showed in Table 2.

In the creation of the simulation environment, it is needed to change the value of parameters for each simulation to reproduce the target scenario. Since it takes high precision and accuracy in dislocating obstacles and walls, changing the value of all three parameters each time would require a massive amount of time. Thus, it is more comfortable to change one parameter value at a time. In particular, the parameter

whose modifications are easier to carry out is going to be varied more often than the other two. Such parameter is the width of the pass, for which changing the position of walls is enough. The most time-consuming modification is the one involving the values of the frequency of static obstacle. Indeed, changing the value of this parameter implies adding one or more obstacles and, consequently, dislocating all the pre-existing ones each time. This process requires a high amount of time and, for that reason, the frequency of static obstacles is kept unchanged as long as possible, modifying its value only twice throughout the whole simulation process.
For such parameter, as shown in Figure 18, it is important to make a specification. Indeed, unlike the isolated parameter scenario, in this case there is a curve in the middle of the path. Since the robot runs into the curve, it is unrealistic to position an obstacle just after or just before the curve and therefore the curve itself is going to replace one of the obstacles. Consequently, if the frequency of 5 meters/obstacle implied positioning 16 obstacles in the isolated parameter scenario, in this situation the obstacles are going to be 15, due to the presence of the curve.

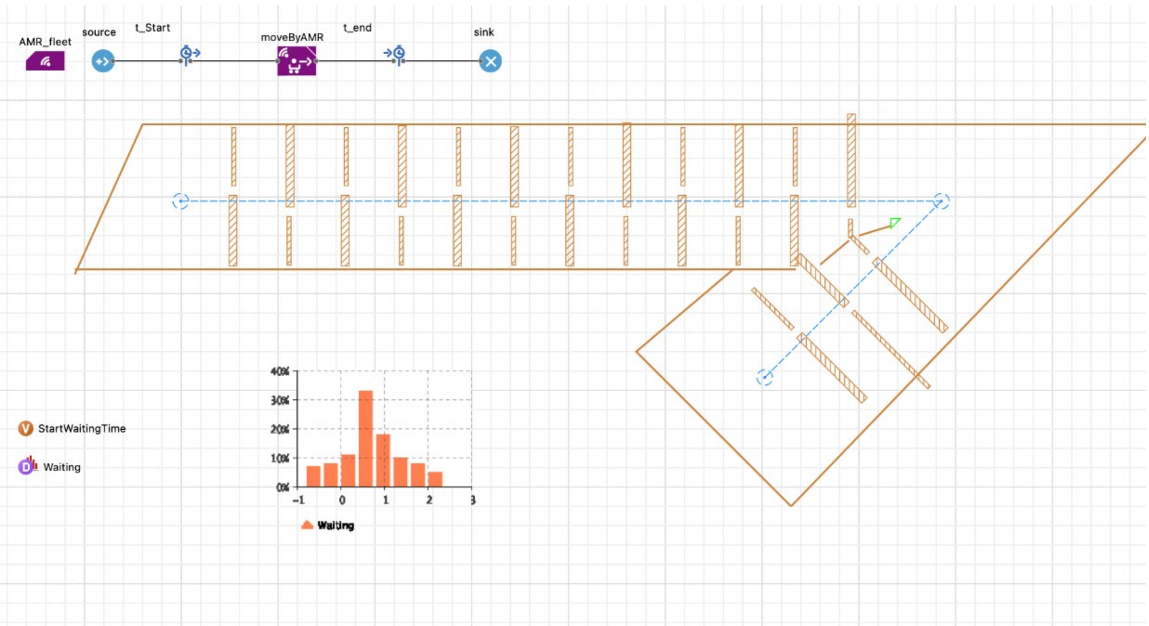


Figure 18. The simulation environment with combined parameters. In this case, the environment consist of a 110-meter long aisle in which many obstacles are positioned 5 meters distant from each other. The width of the pass is kept to 1 meter and the curvature angle equals 45°.

The curvature angle is quite time-consuming because the aisles cannot only be rotated. Indeed, as specified in Chapter 6.4, the curvature angle implies difficulties in terms of actual length of the path, because the AMR tend to cut through the curve

and get as close as possible to the walls. Therefore, it is necessary to move the destination point a little further to make sure the robot travels the same distance as in the straight angle scenario. The additional distance must be calculated, and this makes the curvature angle the second most time-consuming modification to the simulation environment. The simulation that must be carried out is designed exactly the same way as the ones involving isolated parameters: the two variables that allow to keep track of time and return the mean value are set, meaning that the measured time refers to the travel of the robot back to its starting point, right after it unloads the item to the destination point and start traveling its way back to the starting point. The length of the AMR path is kept to 100 meters.
Once the dataset including all the combination of the three parameters is created and the AnyLogic environment is set, it is possible to run the simulation.
The results returned by the simulation are resumed in the following graph. On the X-axis it is possible to see the combination of values that each column refers to.

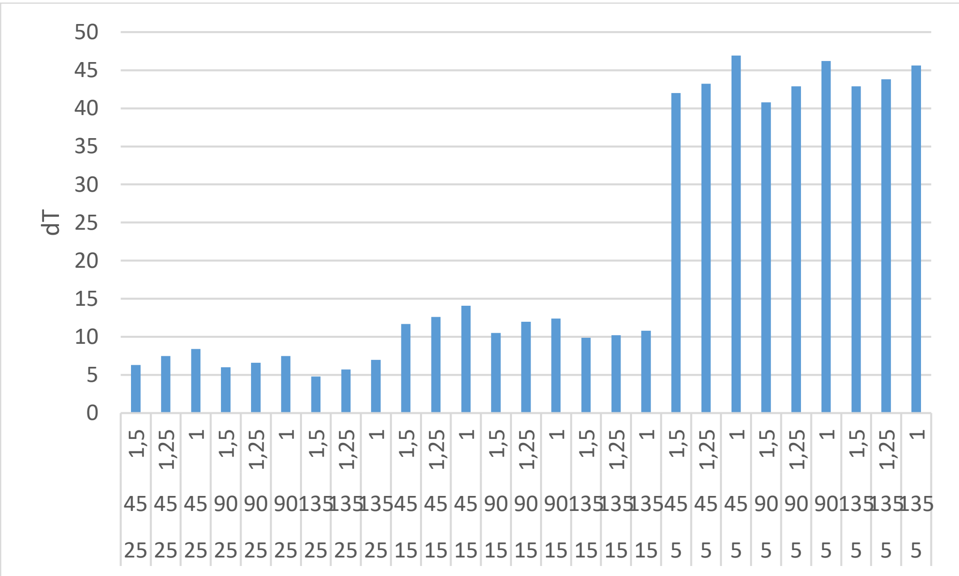


Figure 19. The results of the simulations with combined parameters.

Having said that, it is important to make a further reflection. The results that were just obtained from the simulations with AnyLogic highlighted the time loss under significant circumstances that can reproduce realistic scenario. Nevertheless, the aim of this part of the work is not to associate a mission time loss to each reproducible scenario, which would require a huge amount of time due to the number of possible combinations of parameters. On the contrary, the focus of this study is to investigate and possibly detect an existing relation between parameters or, even more significantly, proofs of what parameter is more influent on the robot mission time. The analysis carried out in the next paragraph allows to state which parameter is key to the reduction of the robot mission time and therefore needs to be avoided in a real-life situation such as a new material handling system implementation. For instance, if the frequency of static obstacles turns out to be the most influent parameter on the robot mission time loss, this study can provide useful information to the designing engineer, who should make a further effort in order to reduce the possible occurring of situations in which multiple obstacles are in the robot path with high frequency.

The process that implies such reflections is pivoted on the statistical analysis called ANOVA, which is explained and executed in the next paragraph and is key to the model implementation with combined parameters.

14.MODEL IMPLEMENTATION WITH COMBINED PARAMETERS.

14.1 ANOVA TEST AND ASSUMPTIONS

The models that were implemented in Chapter 7 must be considered as standards for the implementation of a new model, which enables considering multiple parameters influencing the robot mission time simultaneously. The simulation results are organized in an Excel worksheet and then taken as input data to carry out an ANOVA test, which stands for Analysis of Variance. The data analysis software used for this process is Minitab, a statistics software package that allows companies of any kind to spot trends, solve problems and to process business data in general. It also allows to execute the ANOVA test and give out precise and reliable results.

The Generalized Linear Model of the Analysis of Variance is a regression model and, thus, it aims to detect a potentially existing relation between a dependent variable and an independent one. Based on the number of independent variables, this test is called One-Way or Two-Way ANOVA. Going into detail, a one-way ANOVA is a type of statistical test that compares the variance in the group means within a sample whilst considering only one independent variable or factor. For instance, if it is needed to detect a possible relation between the age of an industrial machinery and the frequency of malfunctions, a one-way ANOVA is required. If it is necessary to state if a relation exists between the age of the machinery, its weight and the frequency with which malfunctions occur, then a two-way ANOVA is required. In this work, since there are three parameters to be considered, a multi-factor analysis of variance must be carried out. The two-way ANOVA can not only determine the main effect of contributions of each independent variable but also identifies if there is a significant interaction effect between the independent variables. Concerning the above-mentioned example, a multi-factor ANOVA would consider an additional factor given by the combination of the machinery age and weight, assessing if there is a cause-effect relation between such combined parameter and the frequency of malfunctions. By analyzing the results, it is possible to understand whether one parameter is more influent on the result, which in this work is the robot mission time, and how more it affects it compared to the other parameters.

Since the analysis of variance can be usefully implemented in many scenarios, it can be carried out in a huge range of application fields such as industry, agriculture, medicine, sociology and more.

Nevertheless, before inserting the simulation results as input data in Minitab, it is necessary to specify some assumptions that the ANOVA test has and make sure that the case study reported in this work complies with such assumptions. The above-mentioned conditions that must be fulfilled are the following:

1) *Interval data of the dependent variable*. This assumption suggests that an ANOVA requires the dependent variable, which in this case is the mission time variation, to be of metric measurement level, which is ratio or interval data. Examples of continuous variables are time and weight, respectively measured in hours and kilograms. In this specific case, the mission time variation is clearly an interval data kind of variable. In fact, interval variables have their main feature in the possibility to be measured along a continuum and have a numerical value. Therefore, the dependent variable of this ANOVA test complies with this assumption.
   Moreover, the *independent variables shall be nominal*. If the independent variables are not nominal or ordinal, they need to be grouped before the multi-factor ANOVA can be done. Nominal variables are variables that are described by two or more categories, but which do not have an intrinsic order. For instance, the curvature angle is an independent variable and its value can only be one out of five pre-determined values, which leads to assume that it's a nominal variable. Such analysis applies to the other independent variables of this work.

2) *Normality*. The second assumption of ANOVA requires the dependent variable to reproduce or at least nearly reproduce a normal distribution for each of the multiple combinations of the independent variables. Testing this hypothesis can be done in several ways: it is possible to build a histogram with a normal distribution curve or carry out a goodness of fit test against normal distribution, which is usually the Kolmogorov-Smirnov normality test.

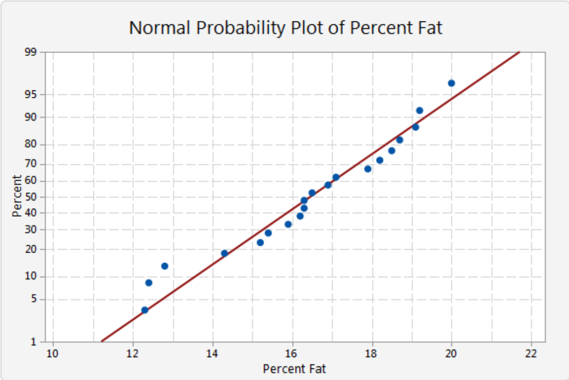


Figure 20. Example of Normality test executed on a population that received a treatment for body fat loss. The closer to the line residuals are, the higher is the probability of residuals following a normal distribution.

The Kolmogorov-Smirnov test, carried out thanks to Minitab statistical analysis software, serves to check if the residuals of the ANOVA test follow a normal distribution. Going more into details, verifying this assumption is all about having or not having enough evidence to prove that the residual are not normally distributed.
The results of the test, as well as a more detailed description of how to interpret its key figures, are reported in the following paragraphs.

3) The ANOVA test assumes *homoscedasticity* of error variances. This simply means that the variance, or error term, of the outcome, which is the dependent variable, is the same across all values of the independent variables. The homoscedasticity test can be carried out through either a Levene's test or a simpler graphical observation of the trend highlighted by the outliers, which represent the variances. In more formal terms, the general rule is that if the ratio of the largest variance to the smallest variance is 1.5 or below, the data is homoscedastic. Graphically, it is possible to assume whether a set of data is homoscedastic or not by reconnecting its trend to one of the two following examples.

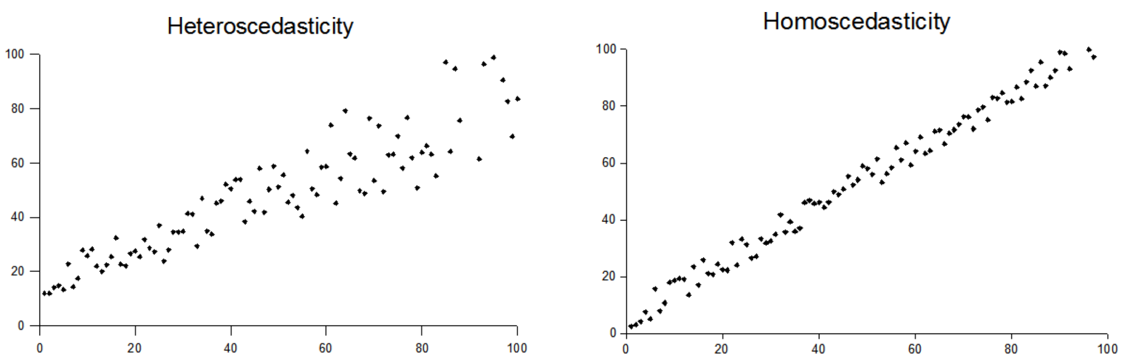


Figure 21. The graphical representation of variances allows to assume the homoscedasticity of a dataset.

4) The fourth assumption, which needs to be verified prior to executing an ANOVA test, is randomness of samples and avoiding *multicollinearity*. Avoiding multicollinearity means making sure that the observations are independent from each other. Since the factorial ANOVA includes two or more independent variables it is important that the factorial ANOVA model contains little or no Multicollinearity. Multicollinearity occurs, for instance, when the samples have not been chosen randomly and therefore the observations between or within groups are not independent.

   Verifying this condition is easier compared to the previous ones. Indeed, it is necessary a Runs Test to verify that the samples have been chosen randomly. This test consists of a statistical procedure that examines whether a string of data is occurring randomly from a specific distribution. It helps determine the randomness of data and can be done through Minitab. Assessing whether the samples were chosen randomly or not is based on the P-value given in output by this test: if the p-value is less than or equal to the significance level, the decision is to reject the null hypothesis and conclude that the order of the data is not random. The significance level is usually set to 0.05. The results of the Runs Test are reported in the following table.

Descriptive Statistics

			Number of Observations	
N	K		≤ K	> K
27	11.1889		15	12

K= sample mean

Test

Null hypothesis	H_0: The order of the data is random
Alternative hypothesis	H_1: The order of the data is not random

Number of Runs

Observed	Expected	P-Value
14	14.33	0.895

Figure 22. The runs test results.

As it is possible to notice from Figure 22, the Runs Test shows that there is not enough statistical evidence to refuse the null hypothesis, since the number of expected runs is not much different from the observed one and, moreover, the P-value is way greater than the limit value, which is usually set to 0,05. The null hypothesis, or H0, in this case, is assuming the randomness of samples. The P-value being equal to 0,895 implies not being able to confute such hypothesis. Therefore, it is acceptable to assess that the fourth condition is verified in this specific scenario.

Moreover, it is possible to assess the independency of observations because of how the software AnyLogic works. Every time a simulation is run, it is considered by the software as a completely different scenario compared to the other previous simulations and, therefore, it is simulated as a whole new environment, which is chronologically independent from the previous ones.

14.2 MODEL IMPLEMENTATION

Before focusing on the results of the tests and proceeding to the regression function computation, it is necessary to make an introduction on what P-value and F-value represent. Indeed, they are the main parameters that are reported in the results and allow making assumptions on the numeric results.
An ANOVA uses the following null and alternative hypotheses:

- H_0: all group means are equal.
- H_A: at least one group mean is different from the rest.

H_0, also known as *null hypothesis*, represents a scenario in which all means are equal and, therefore, there's not one category that's more impactful on the result than the others.
The F-value and the P-value are the statistic values that best describe the results of an ANOVA test. The following explanation aims to give a short introduction on how to interpret these two values.
The F-value tests the equality of means and its value is significant because it is proportioned to the difference between the groups. Indeed, it is the result of dividing the variation between sample means per the variation within samples taken singularly. Thus, the larger the F-statistic, the greater the evidence that there is a difference between the group means. Nevertheless, assessing this is not enough. Indeed, it is necessary to state whether each group is relevant or not and therefore a minimum F-value, over which a group can be considered affecting the results, is required. Since there is not a fixed value over which a group is considered to be affecting the results, it is necessary to consider the P-value.
The P-value is strictly connected to the F-value, it being directly calculated by a formula including the F-value. Unlike F-value, the higher the P-value, the lower is the influence of that group on the overall result. Indeed, such value represents the probability of mistaking when assuming that the null hypothesis shall be rejected or refuted. Moreover, a conventional value of 0.05 is the threshold value over which

there is not enough statistical evidence to reject the null hypothesis and, consequently, it is not possible to assess that one factor is affecting the mission time variation more than the other two.

To sum it up, if P-value is less than α = .05, we reject the null hypothesis of the ANOVA and conclude that there is a statistically significant difference between the means of the three groups.
Otherwise, if the p-value is not less than α = .05 then we fail to reject the null hypothesis and conclude that we do not have sufficient evidence to say that there is a statistically significant difference between the means of the three groups.

14.3 IMPLEMENTATION RESULTS AND ANALYSIS

After specifying the conditions, the meaning of the main figures and running the simulations given by the combinations of parameters, it is possible to carry out the ANOVA test.
The results of the test are reported in the following tables.

Analysis of Variance

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Freq. obst	2	7334.06	3667.03	4466.54	0.000
Angle	2	8.24	4.12	5.02	0.017
Pass.width	2	32.43	16.21	19.75	0.000
Error	20	16.42	0.82		
Total	26	7391.15			

Figure 23. ANOVA test results.

As Figure 23 shows, the reported results concern the degrees of freedom for each group, the adjusted sum of squares, the adjusted mean squares and the F-value and P-value, which were already discussed about. The degrees of freedom are defined as the number of values that are free to vary in a data set. Usually, the degrees of freedom of a system are the result of subtracting 1 to the total number of items. In this case, since ANOVA follows the above-mentioned since three groups are considered, there are 2 degrees of freedom.
The adjusted mean squares and the adjusted sum of squares are more relevant parameters because the allow more conclusion about which parameter is more influent of the dependent variable. Indeed, these indicators show how much variation of the dependent variable, which is the output of the simulations and corresponds to

the mission time loss, is explained by each parameter. Therefore, it is immediate to notice that the frequency of static obstacle, as predictable, results to be the most influent parameter on the mission time variation. The passage width and the curvature angle are comparable in terms of order of magnitude but their values are extremely smaller than the ones characterizing the frequency of static obstacles.
The F-value, which was described to be proportional to the difference between group means, seems to confirm the observation just explained. Indeed, enough large F-value indicates that the term is significant.
The last value reported in the results table is P-value. Although there is a highly significant difference between the influence of the three parameters on the output, observing the P-values it is possible to assess that there is enough statistical evidence to refute the null hypothesis for all the three of the parameters, concluding that they directly affect the mission time variation. Moreover, regarding the frequency of static obstacles and the width of the passage, the P-value equals 0, which is a particularly relevant result that allows being certain when concluding that such parameters have a strong influence on the time loss. The curvature angle shows a P-value which is greater than 0 but less than the significance level of 0.05. Therefore, the parameter must be considered significant.

Model Summary

S	R-sq	R-sq(adj)	R-sq(pred)
0.906091	99.78%	99.71%	99.60%

Table 4. Model summary: goodness of fit values.

The Model Summary table reports the values indicating the goodness of fit of this model. R-squared is the percentage of variation in the response that is explained by the model. It is calculated as 1 minus the ratio of the error sum of squares, which is the variation that is not explained by model, to the total sum of squares, which is the total variation in the model.
Adjusted R-squared relies to the same meaning as R-squared but in this case it is adjusted for the number of predictors in the model relative to the number of observations. Finally, S represents how far the data values fall from the fitted values. S is measured in the units of the response and the smaller its value is, the better the model describes the response.
In this case, the fitting value can be considered extremely good. Indeed, R-squared and Adjusted R-squared are required to reach at least 95% to assume that a model is

well fitting. This scenario exceeds that value so clearly that R-squared nearly reaches 99.8%. Consequently, it is fair to conclude that the fitting is excellent.

Running an ANOVA test with Minitab makes it possible not only to have an overview on what are the most relevant parameters but also to assess which has the biggest influence on the response within the set of values of each parameter. For instance, if the target is to exclude the most influent value of the passage width parameter to limit the mission time variation, it is possible to detect it by observing the results of its respective P-value in the following table.

Coefficients

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	20.678	0.174	118.58	0.000	
Freq. Obst					
5	23.133	0.247	93.81	0.000	1.33
15	-9.100	0.247	-36.90	0.000	1.33
Angle					
45	0.733	0.247	2.97	0.008	1.33
90	-0.133	0.247	-0.54	0.595	1.33
Pass.width					
1.00	1.422	0.247	5.77	0.000	1.33
1.25	-0.178	0.247	-0.72	0.479	1.33

Table 5. Coefficients table.

In the above-reported table, each coefficient of the Generalized Linear Model is described by the shown indicators. The T-value, which is given by the ratio between the difference in means and the standard error of the difference.

It is possible to use the T-value to determine whether to reject the null hypothesis, that states that the difference in means is 0. The bigger the T-value, the bigger the evidence against the null hypothesis and the stronger the influence on the dependent variable. P-value, which works exactly as the previous observations, clearly highlights that a 90-degree angle is not relevant to the mission time loss, as well as a 1,25-meter wide passage. Again, the reason behind these conclusions is that there is not enough statistical evidence to refute the null hypothesis and state that such coefficients have an influence on the outcome.

The main numerical results of the implementation were discussed in the previous paragraphs and a well-fitting model results from them. Although such activities were certainly important, they were not enough to assess if the model can be validated. What is missing is proving that it verifies every assumption of the ANOVA test. In

particular, since the three out of the four assumptions were already proven to be respected, only the third assumption needs verifying, which assumes that the residuals are normally distributed. The preferred way to prove this is a normality test. Using Minitab to perform an ANOVA test allows testing normality of residuals through a Kolmogorov-Smirnov normality test.
As mentioned in paragraph 8.1, this test involves a null hypothesis H_0, which is that residuals follow a normal distribution and a P-value, indicating whether there is enough evidence to refute such hypothesis. In this case, since ANOVA requires residuals to be normally distributed, it is necessary not to refute the null hypothesis in order for the second assumption to be verified. Therefore, P-value shall be greater than the significance level of 0.05 to accept this model. Moreover, the higher P-value, the higher is the probability that residuals actually follow a normal distribution.

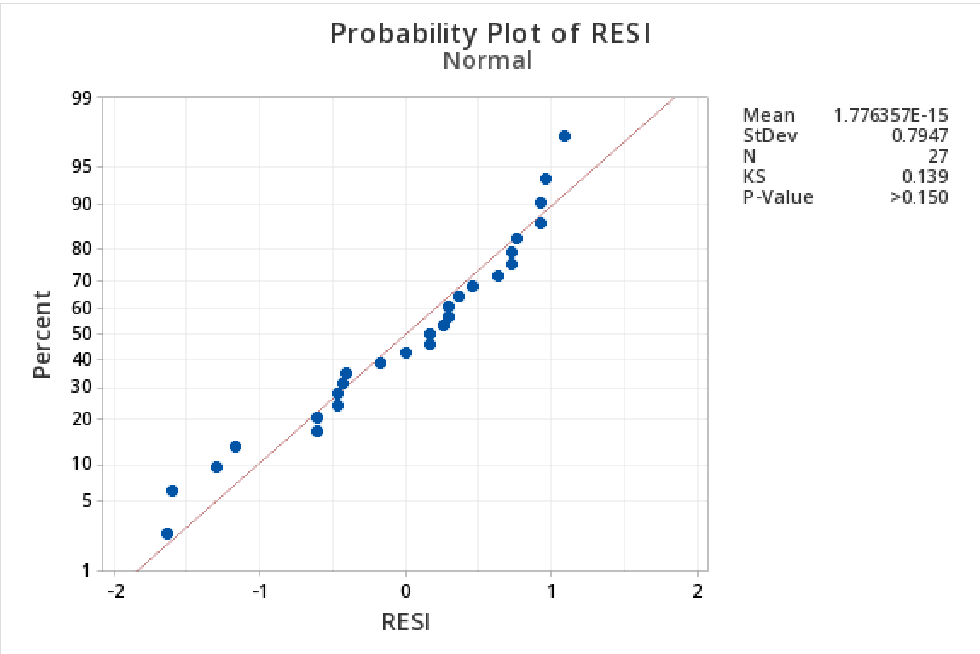


Figure 24. Kolmogorov-Smirnov normality test.

As the graph shows, the P-value given by the Kolmogorov-Smirnov test is greater than 0,150, indicating that there is not enough evidence to certainly deny that residual are normally distributed. Such result indicates that there is a probability greater than 15% of mistaking when assuming that residuals are not normally distributed. Since this probability exceeds 5%, it is therefore not acceptable to refute the null hypothesis and, consequently, the second ANOVA assumption is verified.
Having said that, it is possible conclude that the modal is valid.

For the sake of completeness, Figure 25 reports three additional graphs, let alone the normality test one. The residuals histogram is a representation of the frequency of residuals and it allows perceiving if the residuals are normally distributed from a graphical perspective. Unlike the normal probability plot, this graph doesn't show how far residuals are from a normal distribution and it is therefore the less accurate of the two.

The Residuals versus fits plot is useful to detect whether the model meets the assumption of constant variance. By observing the graph, the focus shall be on detecting a pattern in residuals, which might indicate that variance is not constant and that residuals are uneven spread across the fitted values. Another irregularity that might come up from the graph is the presence of outliers. An outlier is a point which is far from the mean. In this case, even though some point a slightly further from the mean than the others, there is no evidence to conclude that an outlier is there. Moreover, by carrying out an Outlier test on residuals on Minitab, which returned the results reported in Table 6, the previous assessment is confirmed. Indeed, the P-value is way beyond the significance value of 0.05. Therefore, the null hypothesis H_0, which states that all data values come from the same normal population, cannot be refuted.

Last, the Residuals versus Order plot is usually used to verify the assumption that the residuals are independent from one another. Independent residuals show no trends or patterns when displayed in time order. Patterns in the points may indicate that residuals near each other may be correlated, and thus, not independent.

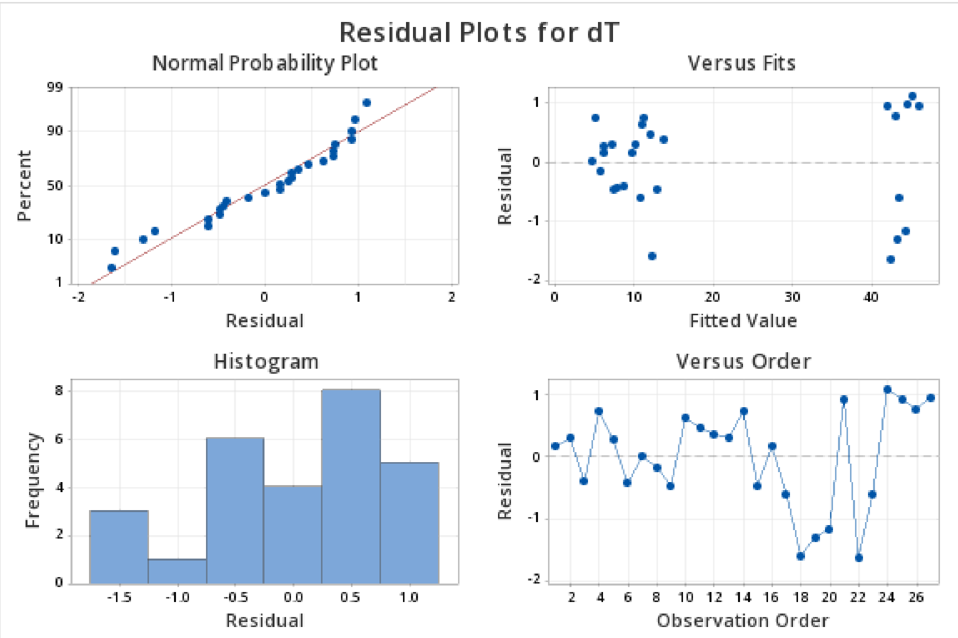


Figure 25. ANOVA summary graphs.

Grubbs' Test

Variable	N	Mean	StDev	Min	Max	G	P
RESI	27	0.000	0.795	-1.633	1.100	2.06	0.899

Table 6. Outliers test.

14.4 FURTHER ANALYSIS AND MODEL IMPROVEMENT

By observing output data, graphs and assuring that all the assumptions for the ANOVA test are respected, it was possible to conclude that the model that was just implemented is valid. Nevertheless, before resuming it all with the final regression equation, it is appropriate to find out whether there are possible improvements to the model that require further investigations.

Basically, improving a fitting model means improving its R^2 or its R^2_{adj}. The following formula, already mentioned in Chapter 7, is the equation for the adjusted R-squared.

$$R^2_{adj} = 1 - \left[\frac{(1 - R^2)(n - 1)}{n - k - 1}\right]$$

Since k is the number of independent variables in the model, R^2_{adj} increases as k increases. On the other hand, removing one factor each time implied a decrease of the resulting R^2_{adj}, as proved by the modifications executed on this model. Such action, though, could only be useful to eliminate a factor whose P-value shows no affection on the mission time or in order to carry out a further analysis or comparison between two isolated variables, but it is not coherent with the purpose of this work, which is considering each parameter that has an influence on the mission time.

Another method to obtain a higher value of R^2_{adj} could be removing outliers. Just as previously mentioned, Grubbs' test did not highlight any outlier and therefore there is no chance of improving the model by getting rid of outliers.

From a different standpoint, improving the model could be interpreted as making it more complete by enlarging the dataset, instead of finding ways to increase the $R^2_{adj.}$ Enlarging the dataset means running more simulations after building new simulation environments with parameter values that are considered relevant for this analysis. In particular, the graph shown in Figure 26 leads to a reflection concerning the gap existing between the ΔT values in correspondence to a frequency of respectively 5 and 15 meters/obstacle.

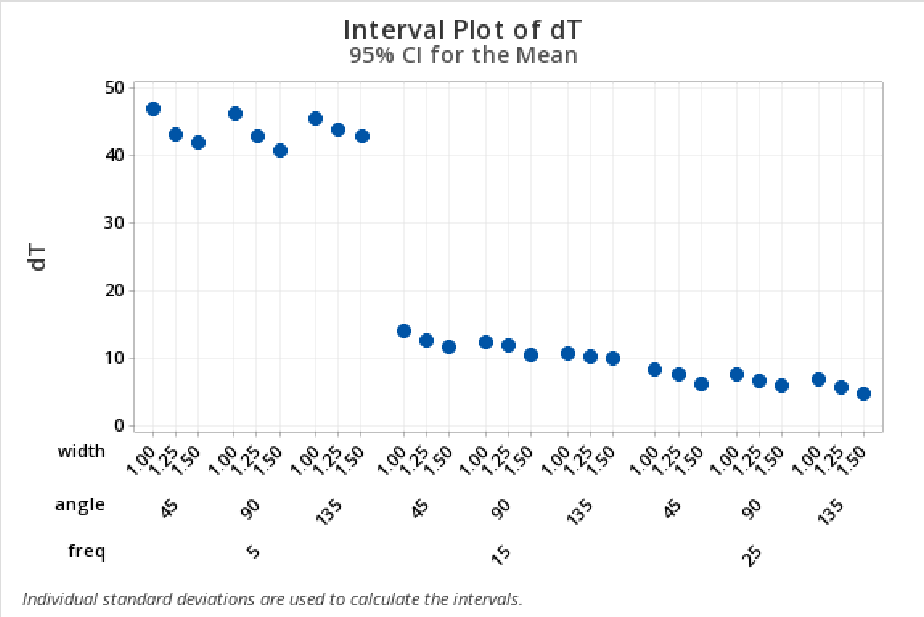


Figure 26. Interval plot of ΔT in correspondence of values of width, frequency and angle reported on the X-axis.

Enlarging the dataset implies using additional values for one or more of the factors involved in this model. While it makes no sense to give the passage width an additional value, and the curvature angle shall be realistically kept to a value not higher than 135°, it is useful to increase the dataset regarding the obstacles frequency instead. Moreover, the graph highlights that there is a huge gap between the ΔT values measured by considering 5 and 15 as values for the frequency of static obstacles. Therefore, it is definitely interesting to analyze how the ΔT trend varies when introducing an additional value of 10 to the frequency of obstacles.

14.5 ANOVA 2.0

Having said that, the simulations are run and the ANOVA test is repeated. The results are the following.

Source	***DF***	***Adj SS***	***Adj MS***	***F-Value***	***P-Value***
Freq. obst	3	7634.06	2544.69	3323.89	0.000
Angle	2	8.28	4.14	5.40	0.010
Pass. width	2	30.26	15.13	19.77	0.000
Error	28	21.44	0.77		
Total	35	7694.04			

Model Summary	**S**	**R-sq**	**R-sq(adj)**	**R-sq(pred)**

0.874972	99.72%	99.65%	99.54%

As shown in Table 7, the P-value indicates great relevance for both the frequency of static obstacles and the passage width, with a value of 0.

The curvature angle turns out to have less relevance than the other two, even though it is still significant since it does not exceed 0,05.

since it is greater than 0,05. Therefore, there is not a significant difference with the results obtained in the previous analysis.

Table 7. Main results of ANOVA 2.0.

The only slight difference is P-value of the curva 7 to 0,010, probably because of the higher completeness of the dataset. Indeed, 36 values were considered in this analysis, rather than the 27 characterizing the previous ANOVA test. From the table regarding the model summary, it is possible to notice that the R^2 and R^2_{adj} indicators are still significantly high, confirming the extreme goodness of the model although they are slightly smaller than the previous ANOVA. Such reduction can be explained by the bigger dataset used to carry out the test. The following table reports the data regarding the single coefficients, with some irrelevant changes due to the enlargement of the dataset. In particular, the 90-degree angle and the 1.25-meter wide passage have smaller P-values, although they are still not small enough to justify stating that such values directly affect the mission time.

Coefficients

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	19.011	0.146	130.37	0.000	
freq					
5	24.800	0.253	98.19	0.000	1.50
10	-5.000	0.253	-19.80	0.000	1.50
15	-7.433	0.253	-29.43	0.000	1.50
angle					
45	0.656	0.206	3.18	0.004	1.33
90	-0.178	0.206	-0.86	0.396	1.33
width					
1.00	1.222	0.206	5.93	0.000	1.33
1.25	-0.236	0.206	-1.14	0.262	1.33

Table 8. Coefficients table for ANOVA 2.0.

It is possible to notice that introducing an additional value of the frequency of static obstacles leads to an increase of the VIF factor from 1,33 to 1,50. Such indicator detects multicollinearity in regression analysis, estimating how much the variance of a regression coefficient is inflated due to multicollinearity in the model. VIF ranges from 1 upwards but the rule of thumb assumes that values of VIF higher than 5 must be taken into serious consideration, which means that the multicollinearity of the model is cause of concern. In this case, its increase on the frequency of static obstacles can be motivated by the increase of variance due to introducing a new value on the dataset of such factor.

The following figure reports the 4 graphs previously discussed, along with the main results of the Kolmogorov-Smirnov normality test, which show that the P-value is still higher than 0,150 and therefore it is not possible to refute the null hypothesis. Consequently, the normality assumption is respected.

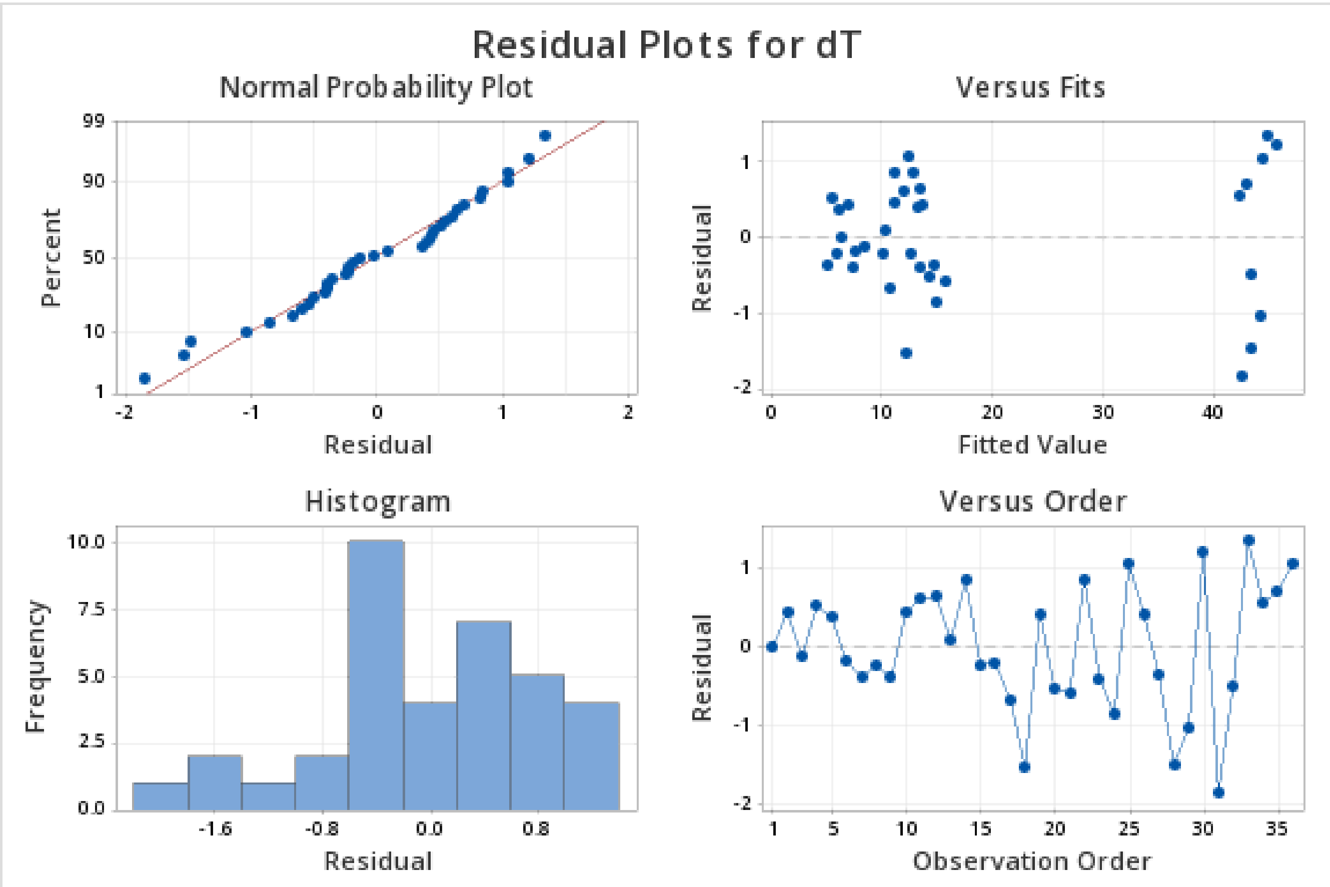


Figure 27. Main plots for ANOVA 2.0.

The Residuals Versus Fits and Residuals Versus Order plots do not show any trend and therefore no abnormality is highlighted. Indeed, residuals seem to be equally distributed around 0.

Another graph that is interesting to report is the Interval Plot of ΔT, in Figure 28.
As predictable from the ANOVA preliminary results, the graph, which now contains an additional portion in the X-axis corresponding to the value that was added to the dataset, has not changed significantly in its trend. Indeed, what results from it is that there is still a huge difference between the mission time loss due to obstacles with a 5-meter distance between them and time loss due to obstacles separated by a bigger distance. Therefore, it is not surprising that the coefficients previously reported in Table 8 show that the coefficient corresponding to the frequency of static obstacles equaling 5 is incomparably bigger than any other coefficient in such table.

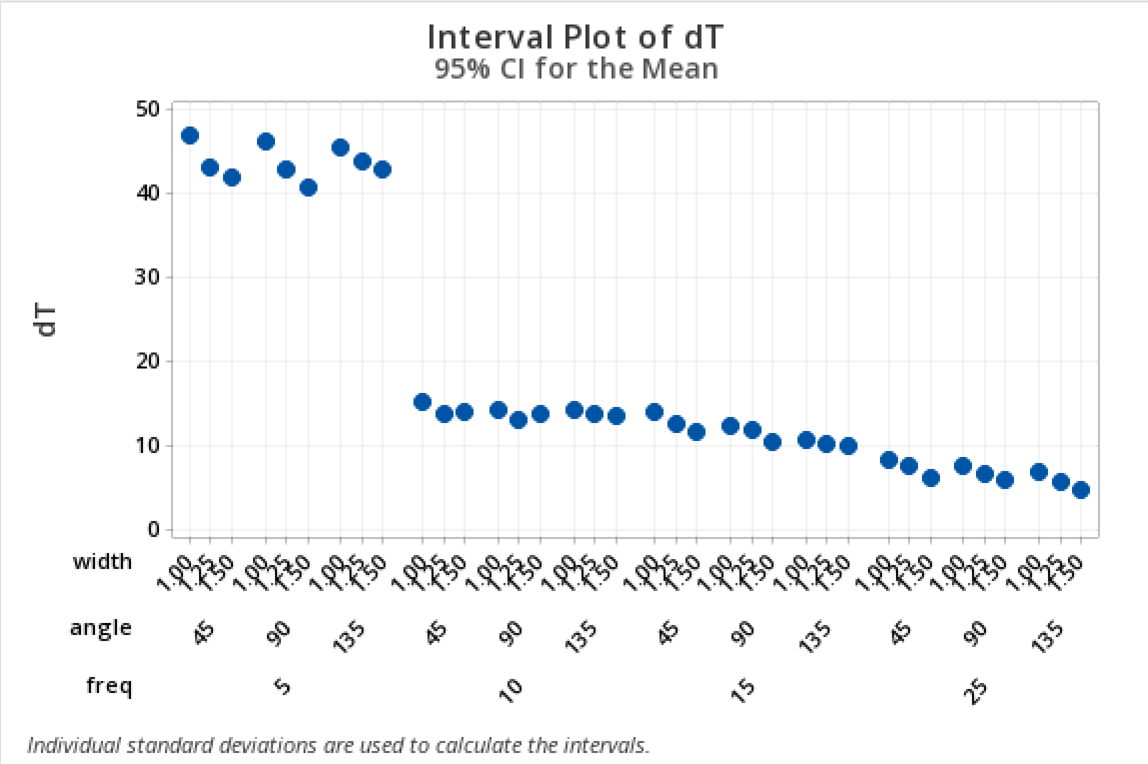


Figure 28. Interval Plot of ΔT for ANOVA 2.0.

The model that was just described respects all the ANOVA assumptions and no outliers were highlighted by Grubb's Outliers test. For this reason, the model can be considered valid. Comparing it to the previous model and to its values of R^2 and R^2_{adj}, ANOVA 2.0 is slightly worse in terms of goodness of fit. On the other hand, this model is resulting from a much more complete dataset, characterized by 36 values instead of 27. It is such greater dataset completeness that makes this model preferable to the other, despite the R^2_{adj} being just under the one of ANOVA 1.0.

Before reporting the regression equation that summarizes the model, a further step is done to bring more completeness to this work and to evaluate a further possibility of model construction.

14.6 MODEL IMPLEMENTATION WITH PYTHON

On a further attempt to obtain a better fitting model out of the given dataset, programming with Python was used to get an alternative regression equation. In this paragraph, the method and the results are shortly presented.

The first step that was taken in the creation of the model was importing the dataset. Then, the input data are rescaled in order to be normalized. After this, a cross validation is applied to check if the model was actually a good fit for the dataset. Having verified such requirement, the model is computed. At first, the model was meant to be linear but the goodness of fit results were not so great. Indeed, the Adjusted R-squared barely reached 68%, which is definitely not enough to be considered a well-fitting representation of the dataset.
Having acknowledged such computational results, it was necessary to increase the degree of the equation, aiming to boost the R^2_{adj} value. Therefore, a second-degree equation was computed and such model turned out to have a significant goodness of fit.
In the reported figure, it is possible to see a small portion of the lines of code used in the making of the model, as well as the mean squared error, the R^2_{adj} and the nine coefficients of the regression equation. It is interesting to specify that this model, just like the previous ones, highlights the bigger impact of the frequency of static obstacles on the mission time loss, compared to the other two factors. Moreover, it is possible to notice that the first reported coefficient, referring to the frequency of static obstacle, has a negative impact on the result but it is positive if it comes to the coefficient referring the same parameter but squared.

```python
[11]: from sklearn.preprocessing import PolynomialFeatures
poly = PolynomialFeatures(degree=2)
X=poly.fit_transform(X)
X_scaled = scaler.fit_transform(X)
#y_scaled = (y-min(y))/(max(y)-min(y))
y_scaled=(y-np.mean(y))/np.std(y)

reg = LinearRegression(fit_intercept=False)

from sklearn.model_selection import KFold,cross_val_score
cv = KFold(n_splits=20, random_state=1, shuffle=True)
scores = cross_val_score(reg, X_scaled, y_scaled,␣
 ↪scoring="neg_mean_squared_error", cv=cv, n_jobs=-1)
print('mean_squared_error: %.3f (%.3f)' % (np.mean(scores), np.std(scores)))

reg.fit(X_scaled, y_scaled)
print(reg.score(X_scaled, y_scaled))
print(reg.coef_)
print(reg.intercept_)
```

```
mean_squared_error: -0.136 (0.090)
0.9187967058003025
[ 0.         -3.47251925 -0.08490034 -0.43387707  2.60547453 -0.0694973
  0.16060329  0.09538455  0.02073401  0.33554767]
0.0
```

Figure 29. Python lines of code for the creation of a fitting model.

Despite the goodness of the above-reported model, its R^2_{adj} equals 91,8% and is therefore still smaller than the second model implemented through Minitab. Therefore, the previous model, ANOVA 2.0 is kept as the best one. Nevertheless, the regression equation is reported for the sake of completeness in the following lines.

$$\Delta T = -3{,}473x - 0{,}085y - 0{,}433z + 2{,}605x^2 + 0{,}095y^2 + 0{,}336z^2 - 0{,}069xy + 0{,}16xz + 0{,}02yz$$

Where:

- x = frequency of static obstacles.
- y = curvature angle.
- z = passage width.

15.FINAL ANOVA 2.0 RESULTS AND MODEL COMPARISONS.

As mentioned in the previous paragraph, the summarizing model this work refers to is the one characterized by the best accuracy, measured by R^2_{adj}, and the best completeness. This is why the regression equation that concludes and summarizes this work is the one resulting from the ANOVA 2.0 test.
In this paragraph a comparison to the model with isolated parameters is carried out, both graphically and through formulas.
The first models, described in Chapter 7 and certainly valid, were obtained by a regression function computation by using MatLab, considering each parameter taken singularly. Therefore, each parameter implies a ΔT trend described by its own formula. The three computed function are summarized in Table 9.

PARAMETER	COMPUTED FUNCTION
Frequency of Static Obstacles (meters/obstacle)	$\Delta T = 6{,}398x^{-\,0{,}2976} - \,1{,}738$
Curvature Angle (degrees)	$\Delta T = 42{,}72\sin(0{,}01841x + 0{,}657) + 39{,}29\sin(0{,}01947x + 3{,}708)$
Passage Width (meters)	$\Delta T = \dfrac{-2{,}354x^2 + 8{,}675x - 6{,}092}{x - 0{,}9454}$

Table 9. Summary of isolated parameters model equations.

The model that was just summarized computes a function for each parameter and does not consider a scenario in which many parameters affect the working environment at the same time. Therefore, to describe more complex scenarios, it is fair to assume that this model simply sums each value of ΔT given by each function. For instance, if an aisle has a 45-degree curvature angle, obstacles 10 meters far from each other and a passage width of 1,5 meters, all the three of the functions are considered and calculated respectively substituting 45, 10 and 1,5 to the “x” variable. Then, results are simply summed to obtain the final time loss.

On the other hand, the regression equation computed by Minitab for ANOVA 2.0 test considers the presence of more than one parameter affecting the environment at the same time, summarizing everything under one function. The formula is the following.

$$\Delta T = 19{,}011 + 24{,}8x_5 - 5x_{10} - 7{,}433x_{15} - 12{,}367x_{25} + 0{,}656y_{45} - 0{,}178y_{90} - 0{,}478y_{135} + 1{,}222z_1 - 0{,}236z_{1.25} - 0{,}986z_{1.5}$$

Where:

- X_i = frequency of static obstacles.
- Y_i = curvature angle.
- Z_i = passage width.

This function is far different from the one reported in Paragraph 9.6. Indeed, the previous one, computed through Python, was meant to allow inserting any value of frequency, angle and width. That function was a generalized one and it allowed calculating the ΔT no matter what values the variables were set to.
In this case, it is important to specify a fundamental difference. The variables that can be observed in the equation are binary. This means that they can only be set to 0 or 1. An example helps understanding how to interpret the above-reported formula.
The variable x_{25} shall be set to 1 only if the frequency of static obstacles equals 25 meters. It must be set to 0 in any other scenario. Therefore, any variable that represents the actual state of the working environment shall be set to 1. Consequently, it is possible to sum all the terms of the equation that are different from 0, obtaining the final value of ΔT. Such equation is more accurate than the one obtain through programming in Python, as the R^2_{adj} witnesses. Despite this, its variables being binary are a limitation to the functionality of the formula, since it is not possible to calculate the ΔT if actual values differ from the ones used in the making of this model. For this reason, and also because of the goodness of fit of both models, both equations can be considered valid and coherent to the purpose of this work.
The following table summarizes the two equations of the combined parameters models.

Minitab Solution	Python Solution
$\Delta T = 19{,}011 + 24{,}8x_5 - 5x_{10} - 7{,}433x_{15} - 12{,}367x_{25} + 0{,}656y_{45} - 0{,}178y_{90} - 0{,}478y_{135} + 1{,}222z_1 - 0{,}236z_{1.25} - 0{,}986z_{1.5}$	$\Delta T = -3{,}473x - 0{,}085y - 0{,}433z + 2{,}605x^2 + 0{,}095y^2 + 0{,}336z^2 - 0{,}069xy + 0{,}16xz + 0{,}02yz$

Table 10. The two equations for combined parameters model.

Having summarized the results of the two models, it is now interesting to compare them. More specifically, it is useful to compare them in terms of the different ΔT output. The comparisons were carried out under two different assumptions. As stated in the previous lines, the isolated parameters models implies adding up the mission time loss caused by each factor. By assuming this, an overestimation of the ΔT value is made, which is because summing the frequency of static obstacles factor to the passage width factor means considering the robot getting past an obstacle twice. Indeed, when considering the frequency of static obstacles ΔT value, multiplied by the number of obstacles in the aisle, an estimation of the total time the AMR takes to overcome such obstacles is already made. It makes little sense to furtherly add the passage width factor for each obstacle and consequently consider each obstacle twice. The results of this approach are far from realistic, since the resulting ΔT is more than twice as big as the ANOVA 2.0 model, which was recognized as the best-fitting one. The graph reported below highlights the huge difference between the two models under this assumption.

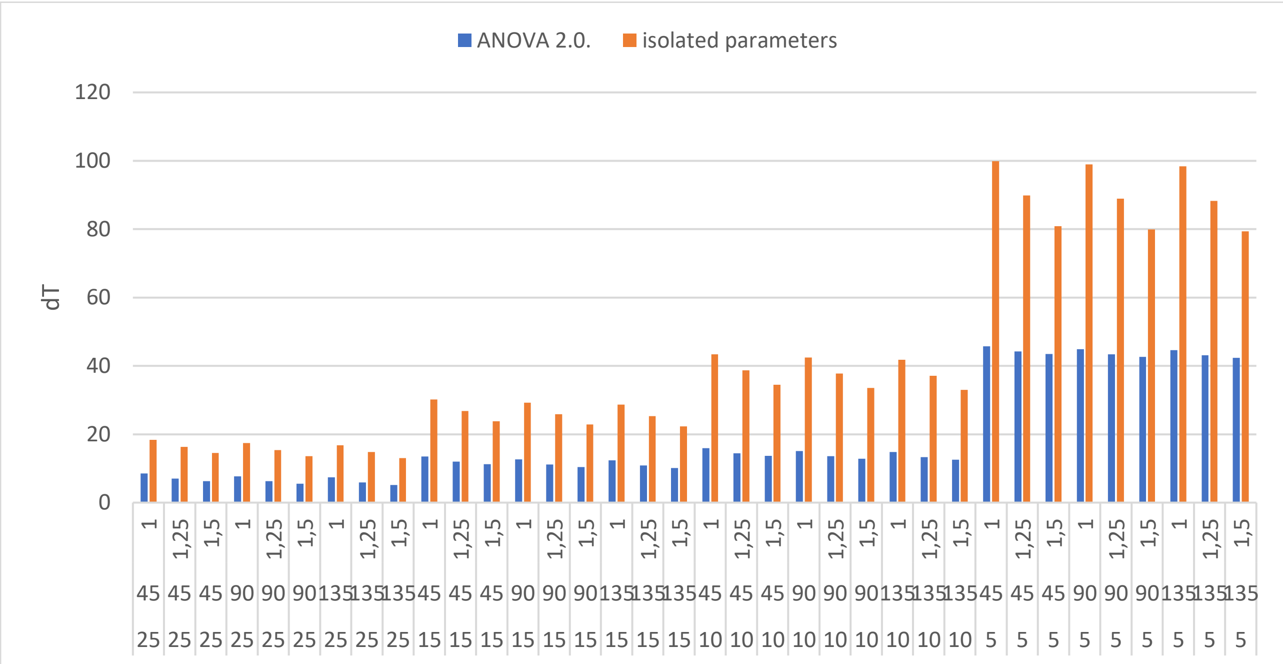


Figure 30. Graph comparing ANOVA 2.0 and isolated parameters models.

A better solution to compare the two models is the following. The only factors that need to be summed are the frequency of static obstacles, multiplied by the number of obstacles in the path, and the curvature angle. By doing this, it is necessary to increase the ΔT by an amount that is proportional to the width of the passage. Therefore, for 1,5-meter wide passages the additional factor shall be bigger than the one corresponding to 1,25 and 1 meter-wide ones. By computing the total time by adding such factors, the results of the two models are comparable. The comparison is reported in Figure 31.

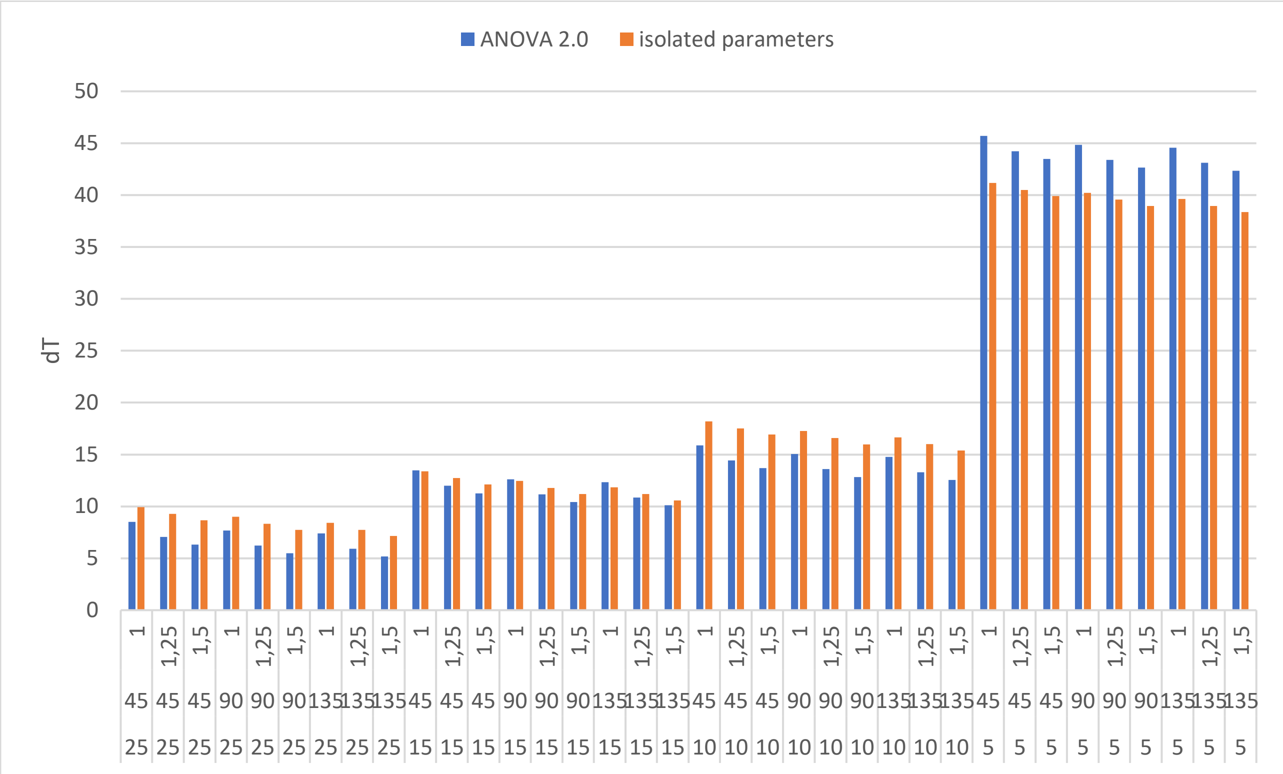


Figure 31. Graph comparing ANOVA 2.0 and isolated parameters models not considering the passage width for each obstacle but only once.

After the above-reported comparison the results analysis is concluded. To sum it up, what was carried out in this phase was using a statistics software to perform an ANOVA test in order to create a regression equation characterizing the model. Starting off from the results of such test, the main trends and interesting information resulting from it were analyzed and discussed. This allowed highlighting what could have been improved or made more complete. Based on this consideration, a further analysis of variance, called ANOVA 2.0, was performed and its results were discussed. Another possibility of creating a model was explored by using Python programming language. All the models created so far were then compared and discussed, after summarizing the whole set of equations. Finally, a comparison of the ΔT resulting from ANOVA 2.0 and the isolated parameters model was made and shown through a graph.

16.LIMITATIONS AND FUTURE WORK

This work finds its functionality validated by the goodness of the models that result from the software analysis but limited by some circumstances at the same time. Although it produced many useful information, it is necessary to admit that a future effort to complete this study is appropriate. As mentioned in the initial chapters, most of the operations involved in the making of this work were performed through AnyLogic simulation software. Let alone the little familiarity with the software, it limited the simulations scenarios because of many reasons. Firstly, the way automated vehicles work in this software do not allow a coherent representation of reality when moving obstacles are involved. Secondly, a huge amount of time was wasted in trying to solve software problems like unusual behaviors of robots. Issues like these resulted in eliminating potential parameters such as the obstacle speed, consequently limiting the field of application of this thesis. Future work is needed to solve such problems; in particular, more experiments in the lab would allow better coherency to reality and a wider set of parameters, compatibly with an acceptable cost raise. Having a bigger set of parameters would allow the creation of additional models involving moving obstacles to cover up a wider range of realistic events.
A further limitation lies in the parameters being categorical. Indeed, when creating a model in which parameters can vary in their value, like previously done by using Python programming language, the model precision considerably decreases, implying a greater percentage of error and uncertainty.

Consequently to these limitations, future work would realistically involve a more frequent use of NTNU Logistics 4.0 Lab or, even better, a real working plant or an industrial warehouse. Experimenting in real working environments would be important to lead trials on larger spaces and longer distances like the ones considered in the simulations of this work. Moreover, it would make possible to have some preliminary feedbacks from workers or engineers on the effective consequences and improvements this work leads to. One last benefit of applying these scenarios to real life situations like a warehouse would allow introducing different kind of moving obstacles, not just a person or another AMR but also any kind of forklift, AGVs or pallet trucks.
Alternatively to leading real life experiments, it would be appropriate to try to repeat the simulations with a different logistics simulation software, like Automod or similar, in order to validate the values of mission time obtained by using AnyLogic. Moreover,

a better completeness could be reached by trying to recreate these models by using different statistical software or other programming applications. By doing this, it would be possible to choose among a wider set of models and to pick the one with the best fitting values.

17.CONCLUSIONS

To sum it up, this work started off with a first phase of literature review, in which every aspect of Autonomous Mobile Robots was discussed: their structure and functionalities, their software and how they represent a new intralogistics method, their pros and cons and their fields of application. As mentioned, such application fields range from industrial plants or warehouses to healthcare solutions as medicine handlers in hospitals. Next, an insight on what caused a variation in the mission time of AMRs was carried out, subsequently analyzing which of those factors could be realistically reproduced in the simulation software and therefore considered as parameters. This phase excluded two out of the six factors that were detected and raised awareness on what is one of the most significant limitations to this work. The next step was the creation of the first model, which involved isolated parameters only, giving as output one function for each parameter. In particular, it is important to highlight that such formulas allow variables to be set to any possible value. The second model was obtained by performing an ANOVA test on the dataset of mission time losses coming from simulations with all the possible combinations of the values of the three parameters. The output was a regression function which only allowed parameters to be set to the simulated values. In the last phase, all the results were shown and compared, highlighting the main differences between the two models.

The aim of this work was providing an initial answer to a topic that was not examined enough in literature: the consequences of AMRs on logistics and intralogistics choices. Indeed, autonomous robots have unique features such as collision avoidance systems and free routing software that represent innovative and incredibly convenient solutions in terms of automation and flexibility. On the other hand, their use implies studying the consequences on the handling time, especially in relation to the surrounding environment and the situations that arise from it. Simulating with combined parameters allowed recreating complex lifelike scenarios and therefore computing a function that quantitatively aims to clear out and quantify the consequences of each scenario on the mission time variation. This work advances a preliminary solution, providing a formula that can be interpreted by engineers in multiple ways. Firstly, it helps engineers in finding different plant layout solutions, adapting it to the situations in which robots perform the best. Moreover, it can be used in a new plant constitution scenario, helping avoiding time consuming situations based on the time loss values related to each parameter.

Most importantly, having a function that estimates the total time loss is useful to determine the actual mission time of a material handling operation. Obviously, knowing this is far from pointless, because it is fundamental in AMR fleet dimensioning problems and therefore it could be critical on the choice of adding or removing one vehicle to the fleet. The nature of such consequences of this would be both logistical and economical.

As mentioned in the previous paragraph, more than one limitation affect this work, which is definitely destined to be reviewed and improved. Furthermore, better versions of the simulation software will be implemented and it will be possible to reproduce more complex scenarios and have a more complete set of models describing lifelike situations. Even Autonomous Mobile Robots, especially in their free routing software solutions and collision avoidance systems, are likely to be brought to a more efficient level, making it necessary to constantly adapt the approach of this work to more evolved working conditions. To conclude, this work provided useful information and statistical evidence, although there is still room for improvement to it, to contribute to the development of such a relevant topic in the intralogistics world.

18.BIBLIOGRAPHY

Draganjac, Ivica, Tamara Petrović, Damjan Miklić, Zdenko Kovačić, and Juraj Oršulić. «Highly-scalable traffic management of autonomous industrial transportation systems». *Robotics and Computer-Integrated Manufacturing* 63, n. C (2020).

Sousa, Norberto, Nuno Oliveira, and Isabel Praça. «A Multi-Agent System for Autonomous Mobile Robot Coordination». arXiv, (2021).

Čech, Martin, Pavel Wicher, Radim Lenort, Tomáš Malčic, Jiří David, David Holman, David Staš andJiří Záruba. «Autonomous Mobile Robot Technology For Supplying Assembly Lines In The Automotive Industry». *Acta logistica* 7, n. 2 (2020): 103–9.

Fragapane, Giuseppe, Hans-Henrik Hvolby, Fabio Sgarbossa, and Jan Ola Strandhagen. «Autonomous Mobile Robots in Sterile Instrument Logistics: An Evaluation of the Material Handling System for a Strategic Fit Framework». *Production Planning & Control*, (2021).

Oyekanlu, Emmanuel, Alexander Smith, Windsor Thomas, Grethel Mulroy, Dave Hitesh, Matthew Ramsey, David Kuhn, et al. «A Review of Recent Advances in Automated Guided Vehicle Technologies: Integration Challenges and Research Areas for 5G-Based Smart Manufacturing Applications». *IEEE Access* 8 (2020): 202312–53.

Fragapane, Giuseppe, René de Koster, Fabio Sgarbossa, and Jan Ola Strandhagen. «Planning and Control of Autonomous Mobile Robots for Intralogistics: Literature Review and Research Agenda». *European Journal of Operational Research* 294, n. 2 (2021): 405–26.

Fragapane, Giuseppe, Dmitry Ivanov, Mirco Peron, Fabio Sgarbossa, and Jan Ola Strandhagen. «Increasing Flexibility and Productivity in Industry 4.0 Production Networks with Autonomous Mobile Robots and Smart Intralogistics». *Annals of Operations Research* 308, n. 1–2 (2022): 125–43.

Ivanov, Dr Dmitry. «Operations and Supply Chain Simulation with AnyLogic», 2017, 97.

Borenstein, J., and Y. Koren. «Histogramic in-motion mapping for mobile robot obstacle avoidance». *IEEE Transactions on Robotics and Automation* 7, n. 4 (1991): 535–39.

Stączek, P., J. Pizoń, W. Danilczuk, and A. Gola. «A Digital Twin Approach for the Improvement of an Autonomous Mobile Robots (AMR's) Operating Environment—A Case Study». *Sensors* 21, n. 23 (2021).

Tang, Niyue. «Securing the future of German manufacturing industry Recommendations for implementing the strategic initiative INDUSTRIE 4.0 Final report of the Industrie 4.0 Working Group».

Enzo Pontarollo. «Editoriale: Industria 4.0: un nuovo approccio alla politica industriale». *L'industria*, n. 3 (2016): 375–82.

Kucukaltan, Berk, Omur Y. Saatcioglu, Zahir Irani, and Okan Tuna. «Gaining Strategic Insights into Logistics 4.0: Expectations and Impacts*». *Production Planning & Control* 33, n. 2–3 (2022): 211–27.

Liaqat, A., W. Hutabarat, D. Tiwari, L. Tinkler, D. Harra, B. Morgan, A. Taylor, T. Lu, and A. Tiwari. «Autonomous Mobile Robots in Manufacturing: Highway Code Development, Simulation, and Testing». *The International Journal of Advanced Manufacturing Technology* 104, n. 9–12 (2019): 4617–28.

Almasri, Marwah, Khaled Elleithy, and Abrar Alajlan. «Sensor Fusion Based Model for Collision Free Mobile Robot Navigation». *Sensors* 16, n. 1 (2015): 24.

Santos, Luis C., Filipe N. Santos, Antonio Valente, Heber Sobreira, Jose Sarmento, and Marcelo Petry. «Collision Avoidance Considering Iterative Bézier Based Approach for Steep Slope Terrains». *IEEE Access* 10 (2022): 25005–15.

Sarmento, José, André Silva Aguiar, Filipe Neves dos Santos, and Armando Jorge Sousa. «Autonomous Robot Visual-Only Guidance in Agriculture Using Vanishing Point Estimation». Cham: Springer International Publishing, 2021.

Cho, Jang-Ho, Dong-Sung Pae, Myo-Taeg Lim, and Tae-Koo Kang. «A Real-Time Obstacle Avoidance Method for Autonomous Vehicles Using an Obstacle-Dependent Gaussian Potential Field». *Journal of Advanced Transportation* 2018 (2018): 1–15.

Chakraborty, D. P., and Tony Svahn. «Estimating the parameters of a model of visual search from ROC data: an alternate method for fitting proper ROC curves». a cura di David J. Manning e Craig K. Abbey, 79660L. Lake Buena Vista, Florida, 2011.

The Concise Encyclopedia of Statistics. New York, NY: Springer New York, 2008.

Everitt, B S, and A Skrondal. «The Cambridge Dictionary of Statistics», s.d., 480.

Vongbunyong, Supachai, Salil Parth Tripathi, Kitti Thamrongaphichartkul, Nitisak Worrasittichai, Aphisit Takutruea, and Teeraya Prayongrak. «Simulation of Autonomous Mobile Robot System for Food Delivery in In-patient Ward with Unity». In *2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, 1–6. Bangkok, Thailand: IEEE, 2020.

Thamrongaphichartkul, Kitti, Nitisak Worrasittichai, Teeraya Prayongrak, and Supachai Vongbunyong. «A Framework of IoT Platform for Autonomous Mobile Robot in Hospital Logistics Applications». In *2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, 1–6. Bangkok, Thailand: IEEE, 2020.

Ramdani, Nacim, Andreas Panayides, Michalis Karamousadakis, Martin Mellado, Rafael Lopez, Christophoros Christophorou, Mohamed Rebiai, Myriam Blouin, Eleftheria Vellidou, and Dimistris Koutsouris. «A Safe, Efficient and Integrated Indoor Robotic Fleet for Logistic Applications in Healthcare and Commercial Spaces: The ENDORSE Concept». In *2019 20th IEEE International Conference on Mobile Data Management (MDM)*, 425–30. Hong Kong, Hong Kong: IEEE, 2019.

Fragapane, Giuseppe, Hans-Henrik Hvolby, Fabio Sgarbossa, and Jan Ola Strandhagen. «Autonomous Mobile Robots in Hospital Logistics. 2020.

Ch'ng, Chee-Henn, Soung-Yue Liew, Chee-Siang Wong, and Boon-Yaik Ooi. «An Efficient Multi-AMR Control Framework for Parcel Sorting Centers». In *2020 IEEE Sensors Applications Symposium (SAS)*, 1–6. Kuala Lumpur, Malaysia: IEEE, 2020.

Faisal, Mohammed, Khalid Al-mutib, Ramdane Hedjar, Hassan Mathkour, Mansour Alsulaiman, and Ebrahim Mattar. «Behavior based Mobile for Mobile Robots Navigation and Obstacle Avoidance». *International Journal of Computers and Applications* 8 (2014).

Sánchez-Ibáñez, José Ricardo, Carlos J. Pérez-del-Pulgar, e Alfonso García-Cerezo. «Path Planning for Autonomous Mobile Robots: A Review». Sensors 21, n. 23 (2021): 7898.

Marques, Francisco, Duarte Gonçalves, José Barata, e Pedro Santana. «Human-Aware Navigation for Autonomous Mobile Robots for Intra-factory Logistics». 2018.

19.SITOGRAPHY

Real Statistics: https://www.real-statistics.com

Support Minitab Statistical Software: https://www.minitab.com/en-us/support/

Statistics How To: https://www.statisticshowto.com

Mobile Industrial Robots – IT: https://www.mobile-industrial-robots.com/en/solutions/robots/mir200/

YouTube: https://www.youtube.com/channel/UC3MRxsNU6AW_hKdExtnGr1Q?view_as=subscriber

MathWorks: https://it.mathworks.com

Gartner, Inc: https://www.gartner.com

AGV Network: https://www.agvnetwork.com/top-amr-applications

Complete Dissertation by Statistics Solutions: https://www.statisticssolutions.com