Towards the design of an effective and robust multi-robot parcel sorting system

F. Mauro

TU Delft

primevision

Towards the design of an effective and robust parcel-sorting system

By

Francesco Mauro

in partial fulfilment of the requirements for the degree of

Master of Science
in Systems Engineering, Policy Analysis and Management

at the Delft University of Technology,
to be defended publicly on Monday August 28, 2017 at 15:00 PM.

Committee Chair:	Prof. Dr. ir.	A. Verbraeck	TU Delft
First Supervisor:	Dr.	M.E. Warnier	TU Delft
Second Supervisor:	Dr.	J.H.R. van Duin	TU Delft
External Supervisor:	Ir.	B. van Dijk	Prime Vision

Preface

This thesis is the result of a research conducted in collaboration with Prime Vision on the development of an effective and robust multi-robot parcel sorting system. This is also the final project for my System Engineering, Policy Analysis and Management Master at Delft University of Technology. The achievement of these thesis results stem from the formation and advantageous conditions guaranteed by both Prime Vision and TU Delft. This master thesis embodies all my ambition and passion for two topics that are of highly personal interest, being automation and logistics. When I started this project, my knowledge of robotics and simulation was limited. However, for the final study of my master, I wanted to challenge and prove to myself that I could perform complicated technical tasks through determination and curiosity ("*curiosity is the wick in the candle of learning", W. A. Ward, Science of Mind*). This project results in a system that I hope can contribute significantly both scientifically and societally.

I would like to thank all my supervisors who have helped me throughout the development of this project. A special thank goes to my first supervisor, Dr. Martijn Warnier, for his outstanding supervision and support. It was definitely an honour to collaborate with you in this project and I believe we have formed a partnership even stronger than a leader with its followers. A special thank also to my external supervisor, Dr. Bernd van Dijk for his immense trust and his life lessons. I also thank my second supervisor, Dr. Ron van Duin, for his enthusiasm, his suggestions related to logistics and for having displayed the model during the Mobility Matters conference in Rotterdam. I would also like to thank my Committee Chair, Dr.ir. Alexander Verbraeck, for his valuable insights into autonomous systems, simulation and logistics. My great esteem for your achievements pushed me to be always concentrated and sharp when arguing my choices. Another special thank goes to the experts who validated my model, which have strongly contributed to the achievement of credible results, thanks to their noteworthy observations. Other thanks to Sjaak Koomen and Thijs Gloudemans for their valuable suggestions.

Finally, I want to thank my parents for having supported me throughout my academic career. I would like to thank my mother that has rejoiced for my achievements and supported me during difficult times. I want to express my gratitude to my father who has always been my mentor. A very special thank to my (best friend and) brother Mattia for always backing and encouraging me throughout my lifetime. I would also like to express my gratitude to Melissa who was always standing by me during this educational path and personal growth. Other thanks go to my friends and my housemate who have contributed to this project with their inputs and by taking my mind off my studies.

"Success is the result of the degree of effort and tolerance, a person dedicates to its work. If you work hard and be patient, results will follow" (my father Mario Mauro)

F.Mauro, Delft, August 2017

Table of Content

Table of Figures

Table of Tables

Executive Summary

In recent years, the postal market has experienced turbulent times. In fact, the demand of parcels has remarkably increased due to the exponential growth of e-commerce. So far, this growing demand has been handled through an amplification of the supply capacities and the purchase of new sorting equipment. Clearly, this solution is neither sustainable nor future-proof. However, the ability of postal operators to plan alternative strategies is constrained by the innermost characteristics of conventional sorting machines, which do not provide enough flexibility (i.e. ability to work at different volume rates), scalability (i.e. ability to scale up or down the system) and adaptability (i.e. ability to reuse the system in other workplaces). Artificial intelligence and robotics are deemed the next game-changers in the field of logistics. In the wake of the new trends in automation and logistics, warehouse operators are reinventing their way of thinking about automation, shifting from static towards more flexible-oriented technologies.

In the Netherlands, PostNL and Prime Vision are working together on the development of a new sorting system, using a multi-robot approach. This master thesis takes place within this ambitious project. The objective of this dissertation is the design of an effective and robust multi-robot parcel-sorting system, in which robots need to sort and transport both light-low volume and heavy-high volume parcels using cooperative and non-cooperative behaviors. The main research question closely relates to the primary objective of this project; hence, this is formulated as:

"What does an effective and robust multi-robot parcel sorting system design look like in which robots behave in a cooperative and non-cooperative manner?"

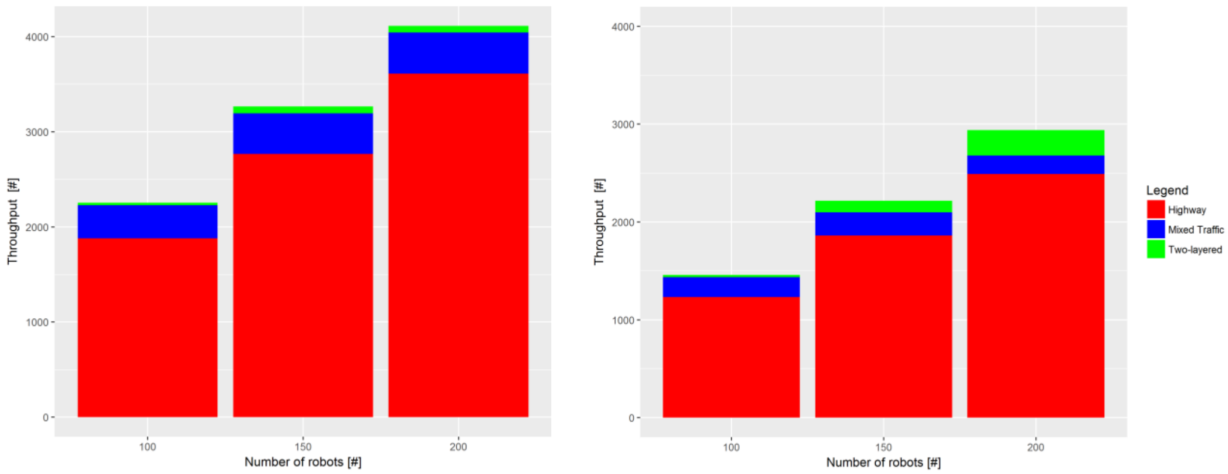
This design project aims at developing a system able to perform basic sorting tasks, like the transportation and appropriate sortation of parcels with different weight and size. Further, this system needs to show off specific qualities desired by postal operators, such as flexibility, adaptability and scalability. The compliance with the Service Level Agreement (SLA) is the main constraint for the design of the new system, which forces postal operators to sort and transport a minimum percentage of parcels per day. This constraint entails the fulfilment of a pre-defined throughput (parcels/hour), which under the specific analyzed input values is of between 5000-8000 parcels per hour.

From the analysis of the literature, the knowledge gap is identified in the concurrent execution of weakly-cooperative (or ST-SR-IA) and strongly-cooperative (or ST-MR-IA) tasks within the same domain of application, using homogenous robots and without sequential task assignment. For the transportation of heavy and high volume parcels, we propose a combined leader-follower and auction-like algorithm, in which the robot that first identifies the task becomes the leader and it starts searching and recruiting followers, based on their internal states. In order to execute concurrently the transportation of light and low volume parcels, robots switch dynamically their behavior into a selfish/non-cooperative way of acting. By doing so, robots are able to react actively according to the tasks they need to handle.

Furthermore, we have devised three traffic design alternatives to control efficiently the traffic inside the transport field. The first alternative, termed Mixed Traffic, considers the utilization of the same space by robots with and without parcels. The second alternative, termed Highway, considers the segregation of robots with parcels and without parcels on the same plane using a fixed path outside the transport field. The third alternative, termed Two-layered, considers the segregation of robots with and without parcels on two different planes.

The sorting operations performed by the new system are modelled and simulated in an agent-based software called NetLogo. Experimental designs are developed by varying input parameters and observing the impact of these alterations on the pre-defined performance indicators. The objective of the experimental designs is the assessment of the impact of cooperative transport on the KPIs, being effectiveness, congestion and fault tolerance. From the analysis of the simulation results on system effectiveness, we can see that the system is able to sort over 8000 parcels per hour in a scenario with 100% light-low volume parcels (best case scenario). Therefore, the system performance enables parcel operators to comply with the SLA constraint. Further, it is clear that cooperative transport brings about a strong decline of throughput. This impact is attributable to the higher usage of resources to transport heavy-high volume parcels. In a scenario with 90% light-low volume parcels (i.e. 10% heavy-high volume parcels) and in Mixed (worst case scenario), the mean of throughput decreases by 55% with 100 robots, 52% with 150 robots and 50% with 200 robots. This negative effect intensifies in Mixed Traffic where the motion of robots with parcels is not separated from the motion of robots without parcels, due to the longer service and return time of robots. From the results obtained from the scenarios with heavy-high volume parcels, it is observed that the performance of robots (i.e.

parcels/robot) rises with the increase of robots in the system. A clear conclusion is that to be effective in scenarios with high percentages of heavy and high volume parcels, the system needs to be fed with higher number of robots.



Figures I.a – I.b: Results from experimental designs on throughput

Service time (i.e. time to transport a parcel from pick-up to drop-off buffers) and distance travelled idle (i.e. distance travelled by robots without parcels) are other two indicators of system effectiveness. From the results, it is apparent how service time increases where the segregation between different robot entities is absent (e.g. in Mixed Traffic), while it is lower in the scenarios that consider the separation of the motion of different robots. In addition, in the Highway, robots travel the longest distance idle, while comparable results are obtained in Mixed Traffic and Two-layered. Therefore, it is evident that Highway offers lower throughput in comparison to the other two traffic configurations due to the long distance travelled by robots without parcels.

From the analysis of results on congestion, it appears that cooperative transport does not influence negatively the degree of congestion, leading to extra workspace requirements. From the results, it is also apparent how the level of congestion is higher in Mixed Traffic, where the motion of different robot entities is not separated. To measure how much time robots take to avoid/dodge other robots, we have identified two types of potential collisions, namely collisions with another robot (type 1) or collisions with formations of robots (type 2). From the analysis of the results, it has emerged that robots can take up to 25 seconds to avoid type 2 collisions, whereas robots can avoid type 1 collisions in a short time (5-9 seconds on average). By multiplying the average number of collisions by the average time spent to avoid collisions, we are able to compute the *conditional congestion* (time spent by robots travelling at a speed inferior to their maximum speed). From the results, we have inferred that in Mixed Traffic, robots spend three times and four times the time spent by robots in Two-layered and Highway to avoid collisions, respectively. Therefore, we can conclude that separating the traffic in the transport field produces higher safety (lower number of interferences) and time saving for collision avoidance.

Finally, from the analysis of results on fault tolerance, the new sorting system demonstrates high robustness, considering that the throughput mean decreases by approximately 5%, even in the presence of five robot failures inside the transport field.

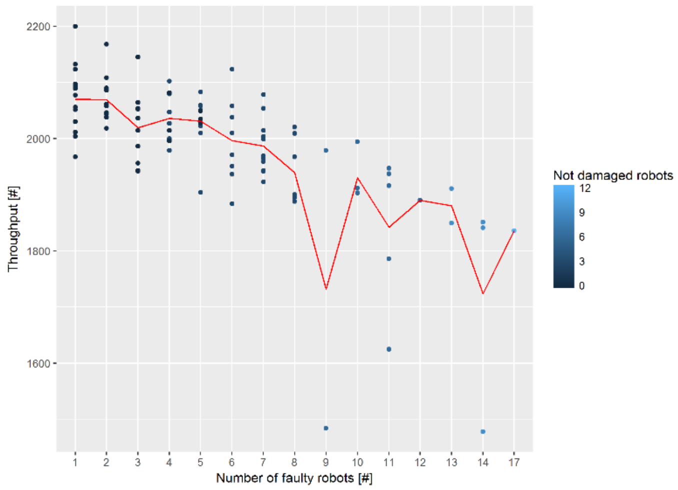


Figure II. Results from experimental designs on fault tolerance

Moreover, the impact of cooperative transport on fault tolerance is revealed. Cooperative transport decreases the robustness of the system by halting undamaged robots in formations in the presence of a failure of one member of a coalition. This problem is addressed by implementing an assistance mechanism that is able to disengage trapped robots in formations. By doing so, the assistance mechanism eliminates the negative impact of cooperative transport. From the results from disruptive scenarios, we have also observed that when pick-up buffers fail, robots need relatively short time to recalculate a new pick-up buffer and reposition themselves across the working buffers. Moreover, when more than one pick-up buffer fails at the same instant, the time to recalculate their position is not different from the event of a failure of a single buffer. In the considered timeframe, after the reconfiguration time, the performance of robots continue to grow steadily.

The results from the experiments show that the new system is able to counteract the problems of traditional sorters, thus making it a suitable alternative for parcel operators.

1 | Introduction

1.1. Effects of e-commerce on postal industry

In recent years, the postal industry has experienced a radical change due to the rise of e-commerce, which has brought about a strong decline of letter mail demand, and a striking increase of parcel volume. From the point of view of postal operators, e-commerce has been like a storm after decades of calm. In 2015, the European Commission declared that online purchasing of goods in Europe was increasing by 22% annually. The B2C e-commerce has offered enormous opportunities to the postal market via the parcel volume growth. In the last years, postal companies have performed remarkably well, increasing their revenues by transporting more both domestically and cross-border (M. Crew et al., 2017). However, e-commerce has also posed many challenges to the postal industry. M. Wen (2004) explains how the development of e-commerce generates fluctuations in productivity and demand. The number of parcels handled every day by postal operators is increasing tremendously in recent years as a result of e-commerce (see Figure 1). In many European countries, the shopping habits are changing extremely fast with an ever-growing percentage of consumers purchasing items online. Furthermore, it is expected that this growth of volumes will not halt in the future, due to a further increase of e-commerce.

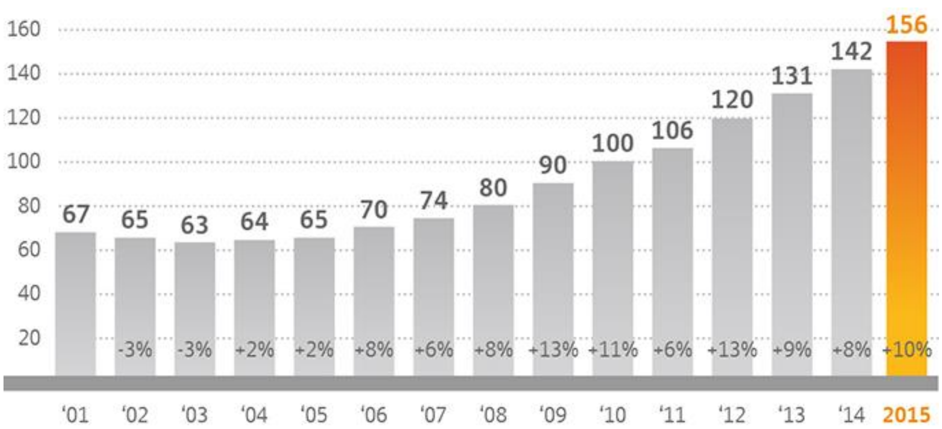


Figure 1: Volume of parcels from 2001 to 2015 in million, retrieved from PostNL Annual Report 2015

These fluctuations of volume require postal companies to be flexible enough to cope with the continuously changing market dynamics. At present, this increase of parcel volume is tackled through an expansion of supply capacities. In this way, postal companies have so far been able to handle the growing demand. Nevertheless, the new expected growth of e-commerce will require more flexibility to satisfy the demand. Continuously purchasing new sorting equipment, which requires large facilities to be installed, is not a sustainable and future proof solution.

1.2. A shift from static to multi-agent automation

At present, the growth of parcel volumes is counteracted by purchasing large fixed equipment, which forces postal operators to open new facilities. In the industry, little has changed in the last decades with regard to the automated systems used for sorting. In fact, the conventional automated sorting systems mostly operated today comprise fixed and large machinery (R.T. Yunardi, 2015). Although the widespread use of these traditional systems, they constrain the ability of industries to plan short- and long-term strategies as a result of their inflexibility and insufficient scalability. This inflexibility is due to the low modularity of the sorting systems that does not enable to re-shape or re-size a warehouse layout and structure based on the fluctuation of volumes. Therefore, in the situation where an industry is called to handle growing volumes, a response can only be to purchase a new automated system or/and moving to another facility. Besides, these systems have low volume flexibility, considering that they are built according to the maximum historical demand. Hence, these systems are not able to handle a demand higher than this maximum demand. This also reduces strongly the utilization rate of traditional sorters. In addition, these systems show inadequate robustness, i.e. they cannot function in the presence of partial failures. This entails that every malfunction of these devices might be leading to serious disruption and delays, affecting the overall operations. Consequently, these systems require an overwhelming amount of maintenance. In addition to the high maintenance cost, the traditional sortation conveyors demand a high initial investment cost (B. Werners and T. Wülfing, 2010).

In 2008, a completely new multi-robot material handling system was deployed, termed Kiva System. Wurman’s revolutionary idea was to coordinate hundreds or thousands of autonomous ground vehicles in the distribution facilities (P.R. Wurman et al., 2008; R. D’Andrea, 2012). In these centers, the driving robots pick up the inventory pods and drive them towards the picking stations where human operators fill the orders. The inventors demonstrate that the Kiva Systems improve speed, accuracy and flexibility of the order fulfillment operations, by turning serial processes into massive parallel processes.

Thanks to the advances in actuators, sensors, advanced control algorithms and machine learning, it is envisioned that, in the years to come, other warehouse operations will be fulfilled using artificial intelligent systems. Thus, the Kiva system can be viewed as the starting point towards the development of new multi-agent systems to perform material-handling operations using a multi-robot approach.

1.3 Research objective

In the Netherlands, PostNL is collaborating with Prime Vision on the development of a new sorting system that can provide a sustainable and future-proof solution against the fluctuations of parcel volumes. By using a multi-robot approach, this system should be able to offer higher flexibility, scalability, fault tolerance and comparable performance in relation to traditional sorters.

Within this ambitious project, this master thesis dissertation plays an important role. The objective of this research is to design and develop a macro-model featuring the operations of multiple robots performing parcel-sorting operations in a sorting hub. In particular, an important requirement for this project is to devise a solution for the transportation of heavy and high volume parcels. In light of this challenge, robots should be able to operate simultaneously cooperative and non-cooperative operations, i.e. operations that require the joint effort of multiple robots and operations that can be performed by single robots, respectively. An important constraint is that robots should be homogenous and of limited dimensions, in order for the system to be economical and for robots to be agile and with reduced energy consumption. This challenge represents the main scientific knowledge gap that we cover in this thesis.

Furthermore, together with Prime Vision, we have defined another objective of this project that is to explore the impact of cooperative transport on system effectiveness, congestion and fault tolerance. Therefore, system effectiveness, congestion and fault tolerance represent the key performance indicators for our multi-robot system. System effectiveness refers to the ability of the system to achieve a specific set of performance requirements, such as number of completed tasks per unit of time. Evidently, this is

one of the most relevant KPIs for the design of a new system. In their research, A. Farinelli et al. (2017) assess the ability of a multi-robot system to maximize system effectiveness, in terms of objects transported per unit of time. Moreover, the authors also point out the importance of diminishing the number of robot interferences in order to minimize the average task completion time. Therefore, congestion is employed as another relevant performance indicator, which is tightly connected with system effectiveness. D. Sun et al. (2014) propose a multi-robot approach to minimize the completion times of transportation tasks. Thus, in their study, the authors only focus on system effectiveness. Z. Yan et al. (2013) underline the importance of analyzing system effectiveness of multi-robot systems, particularly by focusing on qualitative aspects like flexibility, scalability and versatility that provide these systems with potential superior performance. C.S. Kong et al. (2006) indicate congestion as a potential limiting factor for the effectiveness of multi-robot systems. Z. Yan et al. (2013) stress on the importance of devising adequate coordination strategies in multi-robot environments to reduce congestion and increase system safety. L.E. Parker (1995) advocates the inadequate focus of previous work on the issue of fault tolerance, which according to the author, represents a key design issue for real-world multi-robot applications. L. Vig and J.A. Adams (2006) identify fault tolerance as a cardinal issue in multi-robot coalition formation. In their study on multi-robot patrolling, D. Portugal and R.P. Rocha (2013) discuss the negative influence of centralized strategies and global knowledge on fault tolerance.
We can therefore conclude that system effectiveness, congestion and fault tolerance are essential factors to examine for the design of a multi-robot parcel sorting system.

1.3.1 Research questions

The research questions in this thesis relate closely to the primary project objective. Given the role of this project, the main research question can be formulated as follows:

"What does an effective and robust multi-robot parcel sorting system design look like in which robots behave in a cooperative and non-cooperative manner?"

The research sub-questions will facilitate and ease answering the main research question; these are:

1. *How can cooperative and non-cooperative transport be modelled and formalized within the same application?*

2. *How to quantify system effectiveness, congestion and fault tolerance with specific multi-robot performance indicators?*

3. *What design alternatives can be adopted to control the traffic flow of robots inside a sorting hub?*

4. *What is the impact of cooperative transport on system effectiveness and congestion?*

5. *What is the impact of cooperative transport on fault tolerance?*

It is important to mention that sub-questions 2 -3 and 4-5 will be worked out in parallel, as the investigation of one question can provide input for answering the other question. Multiple iterations are needed to tackle these questions. The research sub-questions drive the development of this dissertation, given that taken together, the answers to the research sub-questions lead answering the main research question.

1.3.2 Research Flow Diagram

The Research Flow Diagram (Figure 2) provides a high-level overview of the deliverables and research methods used for answering the research sub-questions. As can be observed, to answer the first research sub-question, we will investigate the literature and use desk research to extract the main algorithms used for the multi-robot cooperative and non-cooperative transport of objects. The main sub-deliverable is the development of a new algorithm for the cooperative and non-cooperative transport based upon the analysis of existing algorithms. In parallel, we execute literature study and experts' interviews to quantify the KPIs (system effectiveness, congestion and fault tolerance), using several multi-robot specific performance indicators. The outcome of this phase serves as input for facing the modeling of the system, where the model is embedded in an agent-based simulation software. In the next phase, following a study of transportation methods to separate traffic of vehicles, we design traffic control methods for the motion planning of robots. Several simulation experiments are designed to evaluate the performance of the system, which is assessed on the basis of the defined indicators. Data analysis methods are then leveraged to inspect the data from the simulation experiments, discover useful information and infer valid conclusions. These analyses allow answering the main research question, indicating how an effective and robust multi-robot system can be designed in which robots are able to dynamically show off cooperative and non-cooperative behaviors.

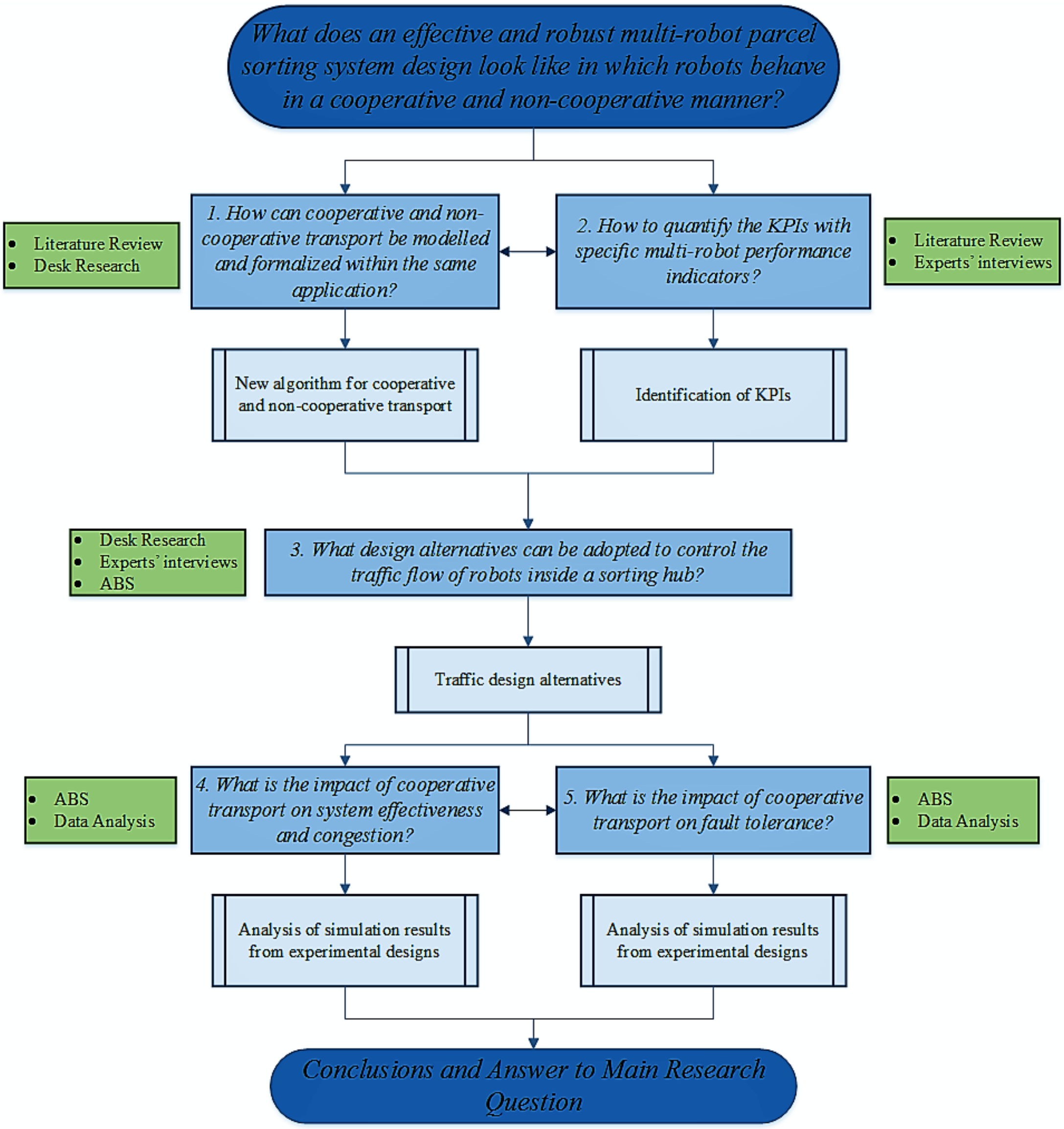


Figure 2: Research Flow Diagram

1.4 Research Methodology

The design of a model featuring an effective and robust sorting system is a systematic and orderly process that requires enacting in accordance with a rigorous methodology. K. Peffers et al. (2007) propose a Design Science Research Methodology (DSRM) for carrying out research based on design science principles (Figure 3). Particularly, this methodology serves as a mental model or structure for the conduct of design researches in information systems. This process model has provided us with guidance throughout the complete design research. It contains six activities ordered in a sequential way with possible process iterations between certain steps:

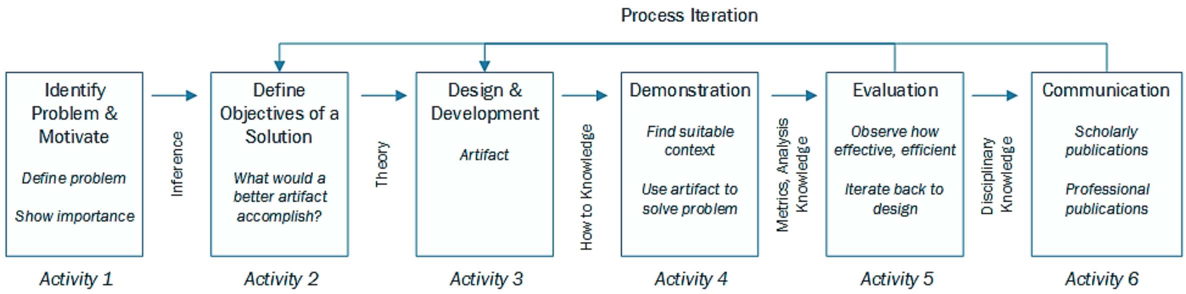


Figure 3: Design Science Research Methodology, retrieved from K. Peffers et al. (2007)

1. *Problem identification and motivation.* The first activity of a design research concerns the identification of a specific research problem and justification of the importance of a solution. In paragraph 1.2, we have shortly delineated the research problem as the insufficient flexibility and scalability provided by traditional sorters to postal operators to oppose the strong fluctuation of parcel volumes brought about by the uptake of e-commerce. However, it is important to first gain a deeper understanding of the parcel practices to discern in more detail the problem to solve and motivate the need for the design of an artifact to counteract this problem. Therefore, in this phase, we need to investigate the state of the problem by analyzing the context in which parcel sorting operations occur and understand the importance of designing a new solution.
2. *Define the objectives of a solution.* When a problem is identified and its solution justified, the problem is converted into system objectives, i.e. requirements for a solution that is able to address the identified problem. The requirements can be functional or non-functional. Functional requirements describe what an artifact is supposed to do, like transport materials, record data, do calculations, make decisions and so on. As S. Robertson (2001) explains, functional requirements lie on the subject manner within the context of the designed system. It is therefore essential to derive these requirements from the observation of parcel sorting practices. Non-functional requirements (also called quality requirements) specify the qualities or attributes we desire a system to have. Therefore, attributes like performance, flexibility, scalability are all non-functional requirements. In order to satisfy the requirements, designers need to consider also the constraints for a

specific system design. Constraints are specific restrictions that influence the way requirements are met, like time, money or performance thresholds (S. Robertson, 2001).

3. *Design and development.* Design and development is the core of design science, which aims at conceptualizing a new artifact able to address the identified problem. K. Peffers defines a design research artifact as "any designed object in which a research contribution is embedded in the design". In this research, the artifact is a (macro-) model featuring the actions of multiple robots performing sorting tasks. Further, as the author advocates, design and development require knowledge of a theory that brings to sustain the solution. Therefore, after the objectives are defined and before designing and developing a solution, this activity needs to be supported by a thorough theoretical knowledge of a theory (in this case of the field of multi-robotics).
4. *Demonstration.* Design and development precede the demonstration activity, in which the developed solution is tested. In our research, this involves the use of simulation to demonstrate how the artifact can be used to solve the problem. Therefore, after we design the macro-model, we implement the model in a simulation software. Further, verification and validation techniques are required to verify the correct implementation of the conceptual model and to determine if the model's outcome is adequate to give insights into the defined problem. This activity is therefore used to build confidence into the last design activity, being evaluation of results.
5. *Evaluation.* In the evaluation activity, computer experiments are conducted and analysis techniques are leveraged to observe and measure the results of the artifact. Depending on the problem and the designed artifact, several evaluation methods can be used. In our case, considering the absence of similar systems, the evaluation of the system is done through the development of several experimental designs and the analysis of results. Moreover, we need to infer comparisons between the new artifact's functionalities and the prior systems and evaluations of the achievement of the system objectives identified in activity 2. At the end of this activity, following the analysis of the results, designers can decide whether to iterate back to activity 3 to improve the design of the artifact or to communicate the achieved results and propose suggestions for further design enhancements.
6. *Communication.* The last activity concerns the communication of the problem, its importance and the final solution to a relevant audience. In this research, communication occurs throughout the development of the design by sharing

information and opinions with several experts, and through the final composition of this master thesis dissertation.

1.4.1 Relation Research Methodology – Research Questions

The research methodology assists us in answering the research questions. The first activity (*Problem Identification and Motivation*) motives the importance of answering the main research question, i.e. designing an effective and robust parcel sorting system. The second activity (*Define the objectives of a solution*) illustrates what we want to achieve through the design of a new system. In particular, this phase introduces the main functionalities of the sorting system, being the transportation and sortation of both small and big parcels. Therefore, when exploring the field on multi-robotics, we need to pay careful consideration of solutions used to solve different tasks. The first sub-research question is answered in activity 3 (*Design and Development*). During the design of the new system, we devise an algorithm to address concurrently tasks that require cooperation of robots and tasks that do not require this cooperation. Before implementing the model into a simulation software in activity 4 (*Demonstration*), we quantify the KPIs, being system effectiveness, congestion and fault tolerance, using specific performance indicators. Therefore, in activity 3, we address sub-questions 1 and 2 in parallel. Activity 4 and 5 (*Evaluation*) are used to evaluate the accuracy of the results of the design system and iterate back to make improvements. In this case, an iteration is made to design alternative traffic control configurations in order to manage profitably the traffic of multiple robots inside the sorting terminal. Therefore, these activities allow us answering the third sub-question. Finally, in activity 5 we develop and execute multiple scenarios to address sub-questions 4 and 5. The analysis of the results from these design experiments shows us how effective and robust the new design system is. Therefore, the execution of all activities of the DSRM enables answering the final main research question, using a systematic design approach.

1.5 Thesis Structure

1.5.1 General overview

The structure of this thesis dissertation is as follows. Chapter 2 provides context by describing the parcel sorting procedures and sorting machines. Chapter 3 investigates the literature on multi-robot systems, with special focus on applications in the logistics field. Therefore, Chapter 2 and 3 build the theoretical foundation upon which we develop our conceptual model. In Chapter 4, we describe all the phases we have followed to build a model of a multi-agent system and implemented it in an agent-based simulation software. Chapter 5 concerns with the development and analysis of results from experimental designs. Finally, Chapter 6 provides answers to research sub-questions and main question; reflections upon the scientific and societal relevance of the outcome of this study; reflections upon the limitations of the methods used and choices made; recommendations for future work for Prime Vision and for possible model extensions.

1.5.2 Relation with the methodology

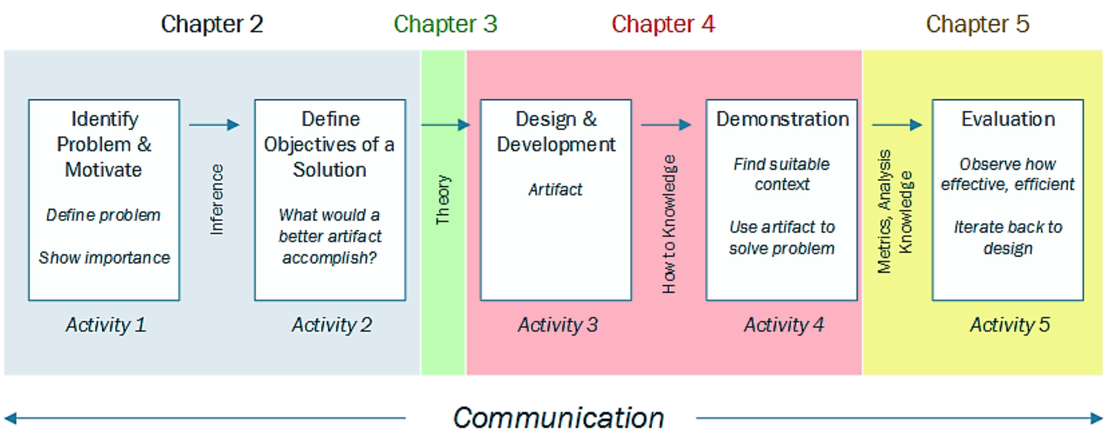
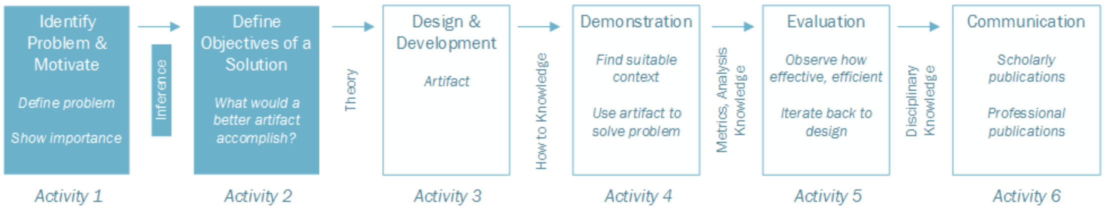


Figure 4: Relation Thesis Structure - Methodology

The thesis structure is based upon the developed research methodology (see Figure 4). Accordingly, Chapter 2 corresponds to the first two activities of the research methodology, i.e. *Identification of problem & Motivation* and *Definition of objectives of a solution*. The goal of this chapter is to describe the context within which the system needs to act, extract the problem and explain the relevance of finding a solution. This leads us towards the definition of the system objectives, i.e. what the system has to do, what attributes should the system possess and what constraints need to be satisfied to meet the project goals. Chapter 3 corresponds to the theoretical foundation upon which the new system is designed. Therefore, the goal of this Chapter is to identify a solution for the identified problem by exploring the field of multi-robotics. Chapter 4 corresponds to activities 3 and 4 of the methodology, i.e. *Design & Development* and *Demonstration*. In this Chapter, the conceptual model is built and implemented in an agent-based software. After the completion of this step, verification and validation techniques are leveraged to ensure the well-being of the model and the results. Here, the goal is therefore to create a valid model that provides an accurate representation of real-life situations. Chapter 5 corresponds to activity 5 of the methodology, i.e. *Evaluation of results.* The goal of this Chapter is therefore to develop computer experiments and infer conclusions from the analysis of the results. Further, inferences are derived from comparisons between results and the system objectives defined in Chapter 1 and comparisons with former sorting machinery. Chapter 6 represents a synthesis of all the Chapters. In this Chapter, we answer the research questions reflecting upon the results achieved, and we provide suggestions for future practical and academic research. As stated earlier, communication is transferred throughout the process via information sharing with experts and, at last, via the composition of this report.

In the next Chapter, we start out the design process by introducing the context, defining and motivating the problem, and finally describing the requirements and constraints the designed system needs to satisfy.

2 | Postal Automation



Introduction

This Chapter represents the first two steps of the research methodology, being the identification of the problem and motivation for its solution (activity 1) and the definition of the objectives of a solution (activity 2). In this Chapter, we explore the field of postal parcel-sorting operations and systems to gain a better understanding of the area under research. From this study, we aim at extracting the root elements of sorting practices and understand the problems related to the use of conventional sorters.

In this Chapter, we consider the parcel sorting process and state-of-the-art parcel sorting machines. The Chapter is structured as follows. In paragraph 2.1, we describe a simplified parcel delivery chain. In paragraph 2.2, we detail some essential definitions that are used in this Chapter and in this thesis dissertation. In paragraph 2.3, we present the current sorting practices that occur in almost every sorting terminal. In paragraph 2.4, we provide a technical description of the traditional parcel-sorting machines. In sub-paragraph 2.4.1, the strengths and weaknesses of traditional sorting systems (i.e. conveyors) are pointed out and encapsulated in a comprehensive table. The weaknesses of traditional sorters represent the problem that suggests the design of a new sorting system. In paragraph 2.5, we describe the functional/non-functional requirements and constraints for the design of a new sorting system. In paragraph 2.6, we illustrate new emerging warehouse automation technologies, with practical examples taken out from real-world applications. Finally, in the conclusions, we emphasize the root elements of sorting practices and the importance of finding alternative solutions to traditional sorters in order to overcome their weaknesses.

2.1 Parcel Delivery Chain

The last decade has witnessed the rise of the parcel delivery industry (PDI), which has become today a significant segment of the transportation and logistics sectors (M.A. Garcia-Romeu et al., 2007). In order to continue growing, the industry needs to remain at the forefront of providing high customer value, which requires exceptionally fast transport and delivery. In addition, cost-savings and reliability are crucial factors for parcel companies, due to the heavy competition in the sector. Cost-savings are achieved by (1) reducing excess inventories, (2) optimizing the scheduling of deliveries, i.e. minimizing the timespan from the unloading of the first parcel until the loading of the last parcel, (3) minimizing transportation costs and (4) minimizing sorting costs (D.L. McWilliams et al., 2005).

Before reaching the customer's door, every parcel undergoes a sequential series of operations. Local delivery vans collect parcels at the shippers and transport them to consolidation terminals, where parcels from different origins are loaded onto outbound trucks. At this point, the loaded trucks deliver the parcels to the central hub terminal (also called sorting terminal). At the central terminal, parcels are unloaded from the trucks at pick-up buffers and routed through a network of conveyors to the appropriate load buffers, where parcels are again loaded onto outbound trucks. Parcels are transported to a segregation or satellite terminal, where they are unloaded from the trucks and loaded onto the local delivery vans. Eventually, the delivery vans take the parcels to the final destination points, typically other businesses (B2B) or customers (B2C).

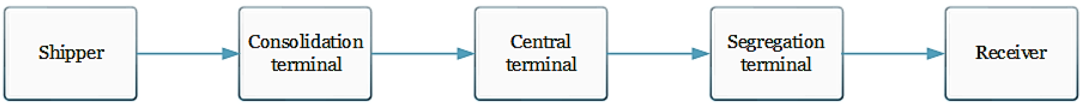


Figure 5: Parcel Delivery Chain

Figure 5 shows the delivery network of parcels from shipper to receiver. However, this is a rather simplistic illustration of a delivery chain. In reality, these chains might present very complex configurations with parcels travelling through several satellite terminals and central terminals before reaching their destinations.

Considering the scope of this research, we only analyze in detail the operations that occur inside a typical central terminal. In particular, we will focus on the automated sorter systems that are the main components of a central terminal. At a hub terminal, inbound trucks deliver shipments from consolidation terminals. The sorting terminal is a central distribution hub where parcels are unloaded and sorted. These hubs differ mostly for their different layout (B. Werners and T. Wülfing, 2010; D.L. McWilliams et al., 2005). However, the procedures performed in these terminals are conventionally the same.

2.2 Relevant Definitions

Before describing the usual practices in sorting hubs, it is convenient to provide some useful definitions of terms that are used in this report (P. McAree et al., 2006):

- ***Parcel*** **–** a parcel is an individual item that arrives at the central terminal in an inbound unit load device. Parcels are discriminated according to their weight, size and addressee. Bar codes contain this information. In particular, parcels can have a max weight of 30 kg and max size of 300 cm (*data provided by PostNL*).
- ***Inbound truck*** – a truck arriving at the sorting hub with unsorted parcels collected at a consolidation terminal.
- ***Outbound truck*** – a truck transporting sorted parcels to their other destinations (e.g. segregation terminal).
- ***Unit Load Device (ULD)*** – an ULD is a container that holds a collection of parcels. These can be separated into inbound ULD and outbound ULD.
- ***Inbound ULD*** – an inbound ULD arrives at a sorting center via inbound trucks. Inbound ULD contains parcels that require sorting.
- ***Outbound ULD*** – an outbound ULD contains parcels that are already sorted and have to be delivered to their other destinations. Outbound ULD is loaded onto outbound trucks and transported to their destination.
- ***Unload (Pick-up) buffer/dock*** – an unload buffer is a location where parcels are unloaded from an inbound ULD and loaded onto a pre-sorter system.
- ***Load (Drop-off) buffer/dock*** – a load buffer is a catchment location where outbound ULD is transported to and loaded onto an outbound truck.
- ***Pre-sorter*** – a pre-sorter is a conveyor equipped with a camera system that reads bar codes. The pre-sorter enables to track the parcels and to avoid their misplacement due to unreadable bar codes. When the parcel is recognized, it is directed to the main sorter system.
- ***Main sorter*** – a main sorter is the main component of the sorting hub. This sorts and transports parcels to the appropriate gravity chutes.

- ***Gravity chute*** – a gravity chute is where sorted parcels drop off from the main sorter and wait to be picked up and placed onto an outbound ULD.

2.3 Parcel Sorting Procedures

An inbound truck transports an ULD from the consolidation terminal to the central hub. Once the inbound ULD arrives at the sort facility, parcels are unloaded at an available unload buffer. Here, workers unload the inbound ULD using forklifts, and they break it down into individual parcels. After the segregation of parcels, these are unified onto single conveyor lines, called pre-sorters. These conveyors identify the parcels by means of a camera system that scans their bar codes. When bar codes are unreadable, the parcels are channeled to the manual stations. In these stations, the logistics operators examine the delivery information and enters it manually (A.N. Tarău et al., 2009).

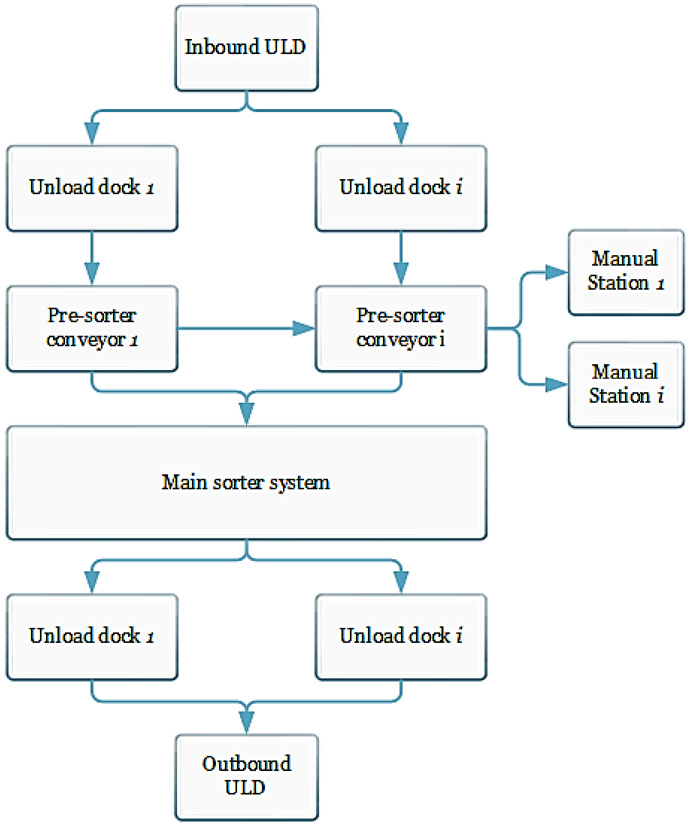


Figure 6: Overview of sorting procedures, retrieved and adapted from K. Fikse et al. (2012)

These unsorted parcels are re-loaded onto the pre-sort conveyor for a second attempt or manually delivered to the appropriate unload buffer. After the parcel recognition is executed, a switch system transfers the parcels to the main conveyor system. The main sorter routes parcels until they reach their appropriate gravity chutes. When the chutes are full, logistics operators load the parcels onto an outbound ULD. Eventually, the outbound ULD is again manually transported to the endpoints, where they are loaded onto outbound trucks. Figure 6 presents an overview of the described sorting procedures.

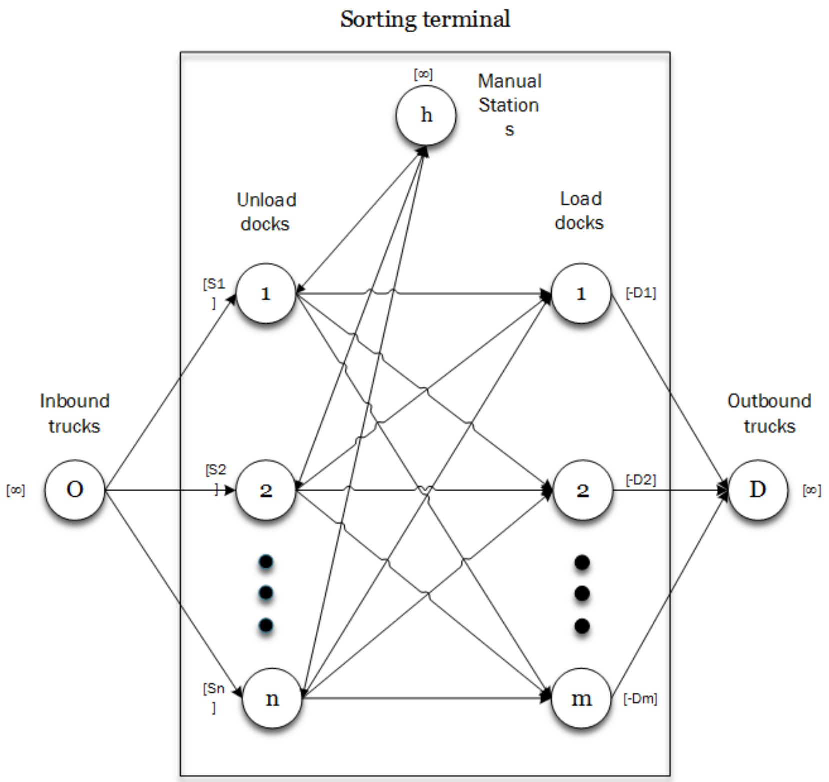


Figure 7: Network flow diagram of sorting terminal, retrieved and adapted from D.L. McWilliams (2005)

In a sorting hub, parcels flow through unidirectional conveyors (arcs) from unload to load buffers. However, bidirectional arcs also exist from unload buffers to manual stations, in case of parcels with unreadable bar codes. The sorting task is therefore a transportation problem from unload buffers to load buffers, through a network of conveyors. Figure 7 presents the network flow diagram of a sorting terminal. In this case, we are only focusing on the sorting system. Therefore, we are neglecting other important resources that constrain the performance of a sorting hub, such as the transfer time to transport the inbound parcels to the unload buffers and the transfer time to transport the outbound parcels to the outbound trucks. Indeed, this research only focuses on the main component of a sorting terminal, i.e. the network of conveyor systems.

This transportation problem is a maximal flow problem, which aims at maximizing the flow from origin node to destination node. Therefore, the objective of this sorting problem is to maximize the parcels flow through the sorting system, i.e. maximize the parcel volume in the outbound trucks (D.L. McWilliams, 2005).

The flow in the network is limited by technical capacity restrictions, which are the limited speed and throughput capacity of the conveyor system. Moreover, the flow is also limited from the physical restrictions of the conveyor systems that can be represented as the distance between the load buffers and unload buffers. These constraints limit the supply capacity, and hence demand capacity, that would otherwise be only dependent on the number of inbound/outbound trucks available at the sorting hubs.

Several authors in the literature have tried to optimize the sorting operations (Figure 8) by acting at layout planning level, destination assignment level, truck scheduling level or conveyor control level (A.N. Tarău et al., 2009; S. Fedtke and N. Boysen, 2014; M.E. Johnson and R.D. Meller, 2002; Y.A. Bozer and H.J. Carlo. 2008).

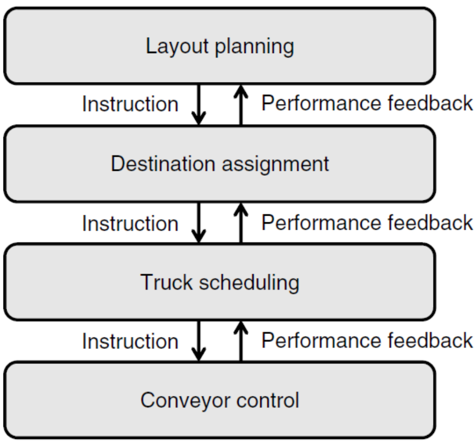


Figure 8: Levels of optimization for sorting procedures, retrieved from S. Fedtke and N. Boysen (2014)

At layout level, optimization techniques are employed to reduce the distances between unload buffers and load buffers, and to choose the appropriate number of buffers. The layout of the central terminal is dependent upon the conveyor systems, which bring along insurmountable physical limitations that constrain the layout choices. Destination assignment refers to the assignment of parcels to load buffers. The objective is to optimize the rapid transshipment of items within the sorting terminal (N. Boysen et al., 2010). The third level of optimization is the truck scheduling. This optimization problem concerns with the optimal scheduling of inbound and outbound trucks at unload and load buffers. Last, the optimization of the conveyor performance, where efficient sorting operations are investigated and developed by means of new control methods and optimization techniques, intended to increment the speed and throughput rate.

Although frequently studied singularly, these levels of optimization are tightly connected and decisions at one level of optimization can affect other levels of optimization. Therefore, a holistic approach is usually employed when studying these operations. This approach enables decision makers to consider the sorting tasks as multi-dimensional and, hence, optimize its subparts, but also taking into account the effect on the overall system. In this project, we mainly focus on the optimization of sorting procedures, thus acting at operational level (last level of optimization).

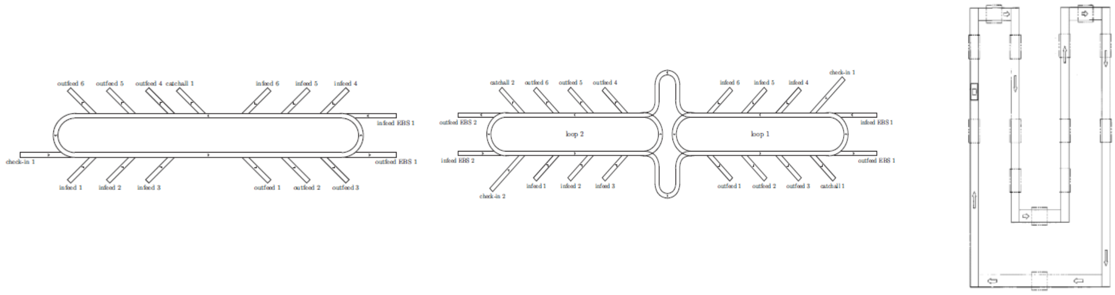
In the next paragraph, we detail some examples of state-of-the-art sorting machines and understand the effects of conveyor systems on the first level of optimization, being layout planning. This will bring us to understand better what the problems of currently used conveyor systems are.

2.4 Traditional parcel-sorting systems

The sorting machines are conveyor systems that transport loads of any shape, size and weight from unload buffers to the right gravity chutes. Conveyor systems have generally two basic types of configuration: line configuration or closed-loop configuration. In the line configuration, materials move unidirectional. On the contrary, in the closed-loop configuration, material travels through a circular track. Sorter systems have a closed-loop configuration, where materials move in a circle until they slip over their appropriate chutes. Within the loop configuration, sorter conveyors can present different layouts. Figures 9a – 9b – 9c display the typical layout configurations of state-of-the-art conveyor systems. In modern sorting hubs, conveyors have either a circular layout (a), a cross-shape layout (b) or a U-shape layout (c) (K. Fikse et al., 2012; B. Werners and T. Wülfing, 2010; D.L. McWilliams et al., 2005, S. Fedtke and N. Boysen, 2014).

Besides their layout configuration, conveyors can be also distinguished for their technical specifications. For parcel sorting, tilt tray and cross belt conveyors are used (R. Bloss, 2013). In *tilt tray conveyors*, every parcel is placed on an individual tray.

Parcels travel along a closed-loop in their trays until they reach their correct gravity chutes. When the correct chute is reached, the tray overturns and slides the parcel into the chute (S. Fedtke and N. Boysen, 2014). *Cross belt conveyors* do not have individual trays for every parcels. A series of conveyor belts drive the parcels towards their chutes.



(a) (b) (c)

Figures 9a-9b-9c: Layouts of typical conveyor systems, (a) circular, (b) cross-shape, (c) U-shape, retrieved from K. Fikse et al. (2012)

All sorting machines have similar characteristics. They have a speed capacity of between 2 and 3 m/s; they are fixed; they have extremely large dimensions; they require high investment and maintenance cost (S. Fedtke and N. Boysen, 2014).

Tables 1a-1b: Squared meters and docking doors for 8,000 parcels per hour (a) and 10,000 parcels per hour (b)

Docking doors [#]	Floor area [m^2]
50	2,910
100	3,250
200	3,920
300	4,595
500	5,985
1000	9,395

(a)

Docking doors [#]	Floor area [m^2]
50	3,090
100	3,380
200	4,050
300	4,720
500	6,115
1000	9,520

(b)

Therefore, the layout of a sorting hub primarily depends on the choice of the conveyor system that needs to be employed. The parcel sorting center is indeed designed with the same shape of the adopted conveyor system. Hence, the sorting terminals can have a rectangular layout (S. Fedtke and N. Boysen, 2014), a cross-shape layout (D.L. McWilliams et al., 2005) or a U-shape layout (B. Werners and T. Wülfing 2010).

The area required by the sorting machine hinges on the throughput capacity required, i.e. number of parcels that need to be handled per hour, and the number of load and unload buffers. The higher the throughput capacity required and the number of buffers, the larger the floor area needs to be. Tables 1a – 1b show the squared meters required by a sorting systems that is capable of handling 8,000 and 10,000 parcels per hour (*data provided by PostNL*). From the data, it is clear that the sorting conveyors conventionally used by parcel operators demand large floor spaces.

However, these technologies have proven to be suitable and efficient when it comes to handle large volumes of parcels every day. Modern sorters have the capacity to process over 15,000 parcels per hour, with very limited missort items (B. Werners and T. Wülfing 2010; K. Fikse et al., 2012). In this way, they are capable of defeating the strong seasonal fluctuation of average quantity of parcels per day in different periods of the year, regularly experienced in sorting terminals (Figure 10). However, the seasonal fluctuation of volumes obligates the companies to purchase a sorting machine that can sustain the maximum peak of volume (typically in December). This entails that the utilization rate of the sorting systems is considerably higher in certain periods and lower in others, with an average utilization rate of around 70%. Daily demand also changes due to the different arrival patterns of inbound trucks.

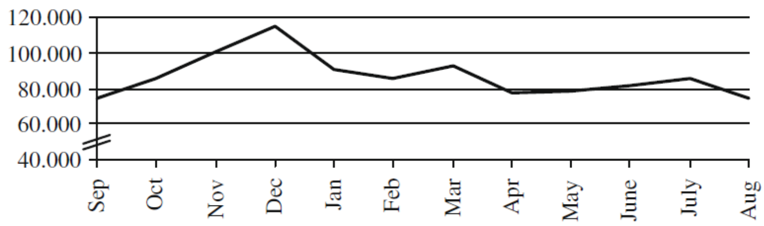


Figure 10: Seasonal fluctuation of parcels volume, retrieved from B. Werners and T. Wülfing (2010)

Consequently, during most part of the season or even of the day, the sorting terminals have unutilized resources. Furthermore, considering the fixed physical structure of these machines, the distances between load and unload buffers can be remarkably long. In the low season, some parcels may need to travel long distances before arriving to their chutes. This reduces the throughput capacity per hour.

Overall, the performance of the sorting conveyors in terms of parcels per hour and speed are satisfactory, according to PostNL. However, these systems bring about some issues that will be investigated in the next paragraph.

2.4.1 Problem identification and motivation

The central terminal uses conveyor systems to transport parcels from unload buffers to load buffers. As stated earlier, these systems provide a remarkably good throughput capacity, which enables parcel operators to resist the anticipated max peak of volume.

However, sorting center responsiveness is increasing, as customers require orders to be delivered the next day. E-commerce is forcing parcel operator to increase their delivery frequency and decreasing order sizes (J. Sadler, 2006). The fierce competition in the parcel market is intensifying the pressure to decrease the operational costs (R. Hamberg and J. Verriet, 2012). Furthermore, the scarcity of warehouse space is leading to serious problems to these companies that need additional space to open other hubs (B.W.F. Angel et al., 2006). These new trends in warehouses have showed up the weaknesses of these systems.

Conveyor systems do not provide the flexibility required by the new warehousing trends. J. Browne et al. (1984) describe two types of flexibility for material handling devices, namely:

- *Volume Flexibility*: the ability to operate a material handling device profitably at different volume rates.
- *Expansion Flexibility*: the ability to expand a system as needed according to the situation, easily and modularly.

Concerning the *volume flexibility*, conveyor systems have a fixed maximum throughput capacity that can sustain. For instance, a parcel company may decide to purchase a system that is capable of processing up to 10,000 parcels per hour. In the event the demand volume rises, a larger throughput capacity may be required. However, these systems are not able to flexibly operate at a volume rate that is larger than initially predicted. Moreover, the *expansion flexibility* is not easily attainable with conveyor systems. Indeed, as stated earlier, material handling devices are large and fixed machinery. Consequently, they cannot be modularly changed, once installed in a warehouse, unless large physical or layout changes are made in the warehouse to accommodate extra equipment parts. However, this would require burdensome investments and time. Evidently, this solution is not sustainable and jeopardizes the objective of reducing operational costs of parcel companies.

Another weakness of these systems is their low *fault tolerance*, which is the ability of a system to keep operating even in presence of failure of one (or more) of its parts.

This is because conveyor systems constitute a single point of failure, which means that a breakdown of a single component of a conveyor system may bring about a standstill of the sorting operations (R. Hamberg and J. Verriet, 2012).

Table 2. Strengths and weaknesses of conveyor systems

Characteristics	State
Volume Flexibility	Low / Average
Expansion Flexibility	None
Fault tolerance	Low / None
Utilization Rate	Low
Reusability	None
Throughput	High
Reliability	High

Furthermore, as reported in the previous Chapter, the *utilization rate* of conveyor systems is generally low in periods where the demand is lower than its peak. Due to the long distances between load and unload buffers, this problem increases the processing time of some parcels and, consequently, it reduces the throughput capacity of the system. Last, these systems are constructed under specific requests of customers and for specific layout configurations. Therefore, these systems are not *reusable* in other applications. This constrains the ability of industries to plan new business strategies.

On the other hand, conveyors have an acclaimed reputation for their *throughput* and *reliability*. In terms of throughput, these systems are able to process tens of thousands parcels per hour, which enables the parcel industry to meet the maximum historical demand. Furthermore, these are highly reliable systems, which means that they are able to work for a long period without any interruption or failure. Table 2 features the weaknesses and strengths of conveyor systems. A postal automation expert has also underlined these strengths and weaknesses of traditional sorters during an interview (see Appendix E for the interview). Therefore, the research problem consists of developing an artifact that can effectively provide a solution to the inadequate (volume and expansion) flexibility, fault tolerance, utilization rate and reusability guaranteed by traditional sorting systems. The development and subsequent dissemination of more flexible technologies would provide parcel operators with an effective solution against the identified shortcomings. Hence, a new artifact might help parcel operators eliminate their fear of coping with undetermined future scenarios. Further, these new technologies would potentially eliminate the need to open up unnecessary sorting terminals, thus rendering this business sustainable and future proof.

Following, we describe the functional/non-functional requirements and constraints that the new sorting system is supposed to have and exhibit. The identification of requirements and constraints (*system objectives*) stimulate the investigation of new design solutions.

2.5 Functional/Non-functional requirements and constraints

The shortcomings of traditional sorters prompt the need to design a new sorting system. As C.L. Dym et al. (2014) point out, “designing new systems is a thoughtful process in which given objectives need to be attained while adhering to specified constraints”. Design objectives, or requirements, are features or behaviors that we wish the system to have or exhibit, while constraints are restrictions on the requirements of the design. Together requirements and constraints form a bounded design framework that help us (the designers) to develop a solution to translate the wishes of postal operators into a real-life artifact.
As already stated in Chapter 1, the requirements can be either functional, i.e. things a system is supposed to do, or non-functional, i.e. attributes we desire a system to have.

Functional requirements. Functional requirements lie on the subject manner within the context of the designed system (S. Robertson, 2001), being in this case the sorting procedures inside the central terminals. Evidently, the basic functional requirement for a sorting system is the transportation of loads from pick-up buffers to appropriate drop-off buffers (e.g. gravity chutes). The sorting system communicates with a camera system, which scans the bar codes, to acquire information regarding the destination of the parcels. Based on this information, every sorting machine routes parcels in a closed loop until the appropriate destination is reached, where parcels slide over gravity chutes. Therefore, any new sorting system needs to perform these two basic functions, namely transport and correct sortation of parcels. As already stated, parcels can be of different weight and size. Therefore, sorters must be able to support different types of parcels; for instance, they need to withstand a given number of kilograms or large-sized parcels. Moreover, sorting systems must provide stability to the parcels they transport. Along their movement, parcels must not slip from the sorting machine.

Table 3. Functional Requirements

System elements	Functional Requirements
Sorting system	▪ Transport parcels ▪ Sort parcels ▪ Support parcels of different weight and size ▪ Provide stability to different parcels ▪ Communicate with camera system to acquire information on destination of parcels ▪ Take parcels with missing information to manual stations ▪ Activate motion if switch mode is on ▪ Deactivate motion if switch mode is off ▪ Generate electric current from wall-outlet or battery system ▪ Convert electrical power to mechanical ▪ Transmit mechanical power to mechanical components ▪ Control data on speed, acceleration, energy consumption, direction and position of parcels
Pick-up buffers	▪ Contain parcels that need to be sorted
Drop-off buffers	▪ Contain parcels that have been already sorted
Human operator or robot surrogate	▪ Pick parcels up from pick-up buffers and place them on sorting system ▪ Pick parcels up from sorting system and place them onto containers ▪ Transport full roll containers to exit gates ▪ Replace full roll containers with empty containers

To support the two top-level functionalities (sort and transport), sorting systems must perform other sub-functions. A sorting system needs to communicate with the camera system to obtain information on the destination of the parcels. When the bar code is unreadable, the sorting systems must bring the parcels to the manual stations where missing information is added. Furthermore, the system needs to possess a switch mechanism in order activate the motion of mechanical components or deactivate its motion at the end of the shift or in dangerous events. To activate the motion of mechanical parts, the system needs to generate electric current from a wall-outlet or battery-based system and convert it into mechanical power. This energy needs to be transmitted to the mechanical parts (e.g. rotors). The system must be also able to control information on electrical current, speed, acceleration, direction and position of parcels. The control of energy consumption is particularly important when the sorting system is powered by battery-based system. To prevent disruptive events, when the battery goes below a certain threshold, the system needs to be promptly recharged. The information concerning the position of parcels is relevant to ensure parcels arrive at appropriate destination and, in the presence of misplaced items, to track and find them.

The overall sorting procedures do not only include the sorting machinery, but also pick-up and drop-off buffers. Pick-up buffers must contain parcels that need to be sorted. A human operator or robot operator (e.g. robot arm) must place continuously parcels on the sorting system. Drop-off buffers must contain parcels that have already been sorted. At these buffers, a human operator, or its robot surrogate, picks parcels up from the sorting system (e.g. from the gravity chutes) and place them onto roll containers. When full, roll containers need to be transported to and parcels loaded onto an outbound truck. A human operator needs to transport full roll containers to the exit points and changed them with empty roll containers.

Table 3 summarizes the functional requirements of the different sorting elements, including the sorting system, pick-up and drop-off buffers and human operators or their robot surrogates.

Non-functional requirements. Non-functional requirements, or quality requirements, are attributes we desire the system to have. Non-functional requirements for a sorting system have already been displayed in Table 2, paragraph 2.4.1. In comparison to conventional sorters, the new system should have high *volume flexibility*, meaning that the system should be able to operate at different volume rates. To achieve high volume flexibility, the system should be designed with the objective of achieving high scalability (or *expansion flexibility*). This implies that the system should have the potential to be increased or reduced in size or scale depending on the amount of work to handle. High scalability also leads to high utilization rate, i.e. low system idleness. Under low demand situations, the number of sorting components should be kept low; while, under high demand situation, the number of sorting component should be increased to improve the system responsiveness.

Furthermore, the system should ideally have high fault tolerance (or robustness), which points to the ability of the system to be able to keep operating in the presence of failures. This entails that the system should not be stopped every time one of more of its components is demoted. Therefore, the system should resist eventual perturbations. The new sorting system should also be reliable, meaning that it should have low system errors. In this research, reliability and robustness mean different concepts, as the former implies that failures should be prevented while the latter implies that in case failures occur, the system should keep functioning properly.
Another desirable attribute the new system should exhibit is the ability to adapt to changing scenarios. The adaptability (or reusability) of the new system could provide

postal operators with the ability to modify the configuration of existing warehouses or to use the system on demand. Further, this attribute should enable the system to fit with current infrastructure. Therefore, the system should not require the purchase of other sorting terminals. The adaptability and scalability of the new system increase the sustainability of new sorting devices compared to fixed sorting apparatus.

Table 4. Non-functional Requirements

Non-functional requirements	Traditional sorters	New sorting system
Volume Flexibility	Low / Average	High
Expansion Flexibility	None	Average / High
Fault tolerance	Low / None	High
Utilization Rate	Low	High
Reusability	None	Average / High
Throughput	High	High
Reliability	High	High
Investment cost	High	Low / Average
Operational cost	Average	Average
Safety	High	High
Predictability	High	High
Ease to operate	High	High
Ease to maintain	High	High

The new sorting system should retain the strengths of traditional sorting devices that offer high throughput. Guaranteeing high throughput is not only a non-functional requirement, but also a constraint for sorting systems. Postal operators have an official commitment with the Government, called Service Level Agreement (SLA), according to which a specific percentage of packages must be delivered to customers within 24 hours. This agreement is a legally binding contract in which the level of service and responsibility are agreed upon between service provider (postal operators) and clients (Government). When the specified level of service (throughput) is not satisfied, postal operators incur penalties (*information received by PostNL and during the interview with a logistics expert; see appendix E*). Therefore, retaining high throughput is a highly significant requirement for the design of a new sorting system.

Another non-functional requirement pertains to the monetary dimension of the new sorting system. We have already argued that conventional sorting systems require high initial investment costs and operational costs, including maintenance costs. To be appealing, the new sorting devices should be economical. The achievement of certain attributes like volume flexibility, scalability and adaptability curtail the high investment cost required to purchase new facilities and new equipment. However, the system should aim to achieve competitive investment cost for new sorting devices and low operational costs (e.g. energy- and maintenance-related costs). In order to decrease the maintenance costs, the system should be also easy to maintain.

In addition, the system should guarantee high safety, especially in the points where human operators interact with the sorting devices. The new sorting system should be easy to block in dangerous circumstances. In order to increase the safety of the sorting operations, the system should be also predictable. When the actions of the sorting devices are well understood by parcel operators, it is easy to anticipate and address unsafe situations. Finally, the system should be easy to operate. Considering that the operations of sorting users are quick and repetitive, humans should not require long time to think and then act.

Table 4 shows the non-functional requirements we desire the new system to exhibit. In this table, we highlight the qualitative score of traditional sorting systems for each non-functional requirement and we show what qualities we want to achieve with the new system. Ideally, the new sorting system should enhance the poor qualities of conventional sorting systems and retain their best qualities, such as throughput and reliability.

Constraints. Constraints are limitations on the features or behaviors of the design. Table 5 incorporates the thresholds for specific parameters (*data provided by PostNL*). The most important constraint, as already illustrated, for postal operators is the compliance with the Service Level Agreement. To comply with this agreement, the sorting devices must be able to withstand a high volume of parcels per hour. A typical threshold inside sorting capacity is 12000 parcels per hour, with 200 sorting directions (destinations) inside a floor area of 7500 m^2. The volume of parcels can range from 1000 to 50000 parcels per hour. The number of sorting directions can range from 2 to 1000 destinations. The floor area can range from 100 to 10000 m^2. In order to comply with the SLA, moreover, the system should not fail in the presence of one or more malfunctions of its components. Traditional sorters do not have this ability and they force postal operators to operate an overwhelmed amount of maintenance to avoid disruptive events. Furthermore, the sorting systems must be able to withstand parcels weighing from 2.5 to 30 kilograms and with size from 20 to 300 cm. Therefore, the new sorting system must be able to convey small and big parcels. The parcel destination distribution is typically uniform across the pick-up buffers, as postal operators strive to have more distributed operations.

Table 5. Design constraints values

Parameter	Unit	Typical Value	Range
Sorting Center capacity	[parcels / hour]	12000	[1000; 50000]
Number of sorting directions (destinations)	[]	200	[2; 1000]
Floor area	[m^2]	7500	[100; 100000]
Unloading operator capacity	[parcels / hour]	800	[10; 1500]
Loading operator capacity	[parcels / hour]	900	[10; 1500]
Loading operator walking speed	[m/s]	1	[0.1; 4]
Loading operator container count	[]	# sorting directions divided by # unloaders	[1; 75]
Container exchange distance	[m]	4	[0.5; 50]
Parcel size distribution	[cm x cm]	50 x 50	[14 x 9; 180 x 80]
Parcel weight distribution	[kg]	2.5	[0; 30]
Parcel destination distribution	[]	uniform	[uniform; non uniform]
Container max number of parcels	[parcels]	30	[5; 250]
Container max loaded volume percentage	[%]	90	[50; 100]

However, some pick-up buffers might have higher or lower supply capacity in certain points in time. The loading and unloading capacity of operators are also limiting factors to the performance of sorting systems. Human operators are typically able to operate 800 unloading operations and 900 loading operations per hour. This implies that they can load one parcel every 4 seconds and retrieve one parcel every 5 seconds. This value can range from 10 to 1500 parcels per hour. Another constraint is the maximum amount of parcels that can be placed onto a container, which depends on the size of the parcels. Typically, the max number of parcels onto containers is equal to 30 parcels. This number can range from 5 to 250 parcels. When containers are full, they need to be transported to the exit gates and replaced with empty containers. The distance between the drop-off buffers and the exit gates is typically 4 meters. Considering a walking speed of 1 m/s, this means that this operation takes around 10 seconds. The number of human operators depends on the number of drop-off buffers required by the system, and it generally ranges from 1 to 75 people.

Another constraint for the new system concerns the cost factor. To be attractive the new system should be cheaper than conventional sorters. The cost for a traditional sorter varies largely, depending on the specific requirements and technical specifications. As already argued, these systems require high investment and operational costs. The new system should therefore aim to reduce these costs. Further, conventional systems are powered by wall-outlet systems. When a new system is battery-powered, the battery life should be high in order to minimize the interruption to replace or recharge the energy of sorting devices.

Following, we describe the current tendency in warehouse automation, providing some examples of companies that are showing the way towards more flexible technologies.

2.6 Emerging warehouse automation technologies

The future of warehouse systems sets upon their flexibility. Warehouse systems do not have to be adjustable to just a set of pre-defined scenarios, but they have to be able to cope with unpredictable circumstances. In recent years, multi autonomous mobile devices have emerged as transportation means in warehouses. These systems have changed the way we look at warehouse automation, as they have altered their warehouses from static networks of fixed machineries to distributed networks of autonomous agents.

In 2008, the first multi-robot system was deployed in a warehouse, termed the “Kiva” system (P.R. Wurman et al., 2008). The Kiva proposes an avant-garde and disruptive approach to material handling, by coordinating hundreds or thousands of autonomous ground robots in distribution centers (P.R. Wurman et al., 2008; R. D’Andrea, 2012; E. Guizzo, 2008). Here, the driving robots pick up the inventory pods and drive them towards the picking stations where human operators fill the orders. Thus, using the Kiva system, workers “do not have to walk over to the shelves to get things, the shelves come to them”. The designers demonstrated that the Kiva system improves productivity, speed, accuracy and flexibility of order-picking operations.

On the scent of the Kiva system, the use of multi-robot systems for logistics operations increased significantly over the last years. Few years ago, Ocado, the world’s largest online grocery retailers, embarked on the development of a new order fulfilment automated technology, which employs thousands of robots. Mounted on a metallic frame, the robots operate on a grid, storing and retrieving bins containing groceries stacked into cells. This system, called the “Hive”, is able to operate large number of robots at extremely high density. Furthermore, the Hive has enabled Ocado to dispose of the need for aisles, providing more space for groceries and improving the overall warehouse performance, by drastically reducing the time to transport the goods from shelves to trucks.

The AutoStore system, by Swisslog, is another tangible example of how controlling multiple robots executing transport and lifting operations is not beyond the bounds of possibility. AutoStore acts very similarly to the Hive, operating above a grid, retrieving the requested bins and transporting them to the picking stations. According to Swisslog, the system provides optimal utilization of spaces, minimal downtime due to the elimination of single point of failures, and high efficiency with up to 500 bins transported to the picking stations per hour (C. Maino, 2014).

Recently, another logistics company, operating in China, called STO Express, has published a video featuring a large number of robots performing sorting operations of small-sized parcels. The robots operate on an elevated grid; transporting parcels from pick up to drop off stations where parcels are slid over into roll-containers situated on the underlying terrain. This system evokes the Kiva system, as it uses bar-coded stickers laid out at circa one meter from each other, in a grid. Using this solution, robots navigate the warehouse with cameras only watching the ground, and not at the surrounding environment. Robots acquire the information with regard to their position from the ground, while a central computer dispatches data to robots regarding the path to follow, the destination cells and the traffic control.

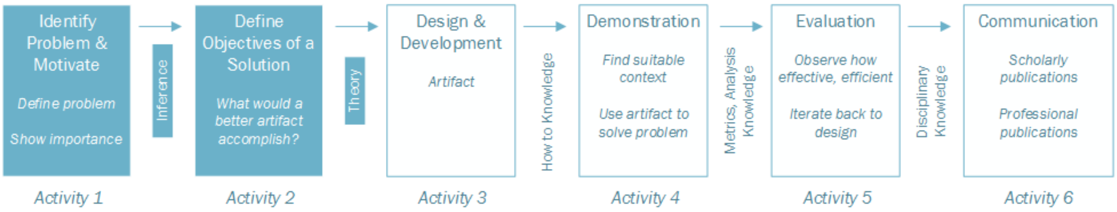
All these distributed warehouse automated solutions have speeded up the transition towards flexible warehouses. However, it is envisioned that, in the years to come, the warehouse operations will become even more flexible by relying even less on the fixed infrastructure, giving more authority to the robots and increasing the adaptability of the systems.

Conclusions

From this Chapter, some conclusions can be derived, which we will be used for the development of a conceptual model of a new sorting system. First, the most important root elements of every sorting hub are the following: number and type of parcels, unload and load buffers, sorting systems (e.g. conveyor belts), Unit Load Devices

(ULD) that can be inbound or outbound, and inbound or outbound trucks. Second, the sorting operations executed in a sorting hub can be represented as a max flow transportation problem, which aims at maximizing the amount of parcels conveyed from unload to load buffers. Third, the sorting machines employed at present may differ with regard to their layouts, but share similar features. These machines are fixed apparatus, which require large warehouse spaces and investments. They have become notorious for their high performance and reliability, but also for their low flexibility (volume and expansion), low fault tolerance (meaning that they have just a single point of failure), low utilization rate and none reusability. Therefore, a new sorting system should be designed with the basic functionalities of every sorting machines, namely transport and sorting loads of different size and weight. Moreover, the new system should exhibit high flexibility, fault tolerance, utilization rate and reusability that represent the weaknesses of traditional sorters. In addition, in order to comply with the Service Level Agreement, the new system should be also able to transport and sort a high volume of parcels and respect specific constraints that are intrinsic to every sorting terminal. Some companies have broken the ground, presenting new automated alternatives to fixed machines, showing the next generation of warehouse technologies. Warehouses are no longer seen as static networks, but as distributed networks of mobile robots. These systems have increased tremendously the flexibility of warehouses; nevertheless, their dependency on rigid infrastructures (e.g. grids or metallic frames) reduces the adaptability of these systems, which can be used only under certain warehouse configurations. With the progress in the fields of robotics, AI and communication, we foresee that a further evolution in warehouse automation is to be expected, with agents operating autonomously logistics operations. In the next Chapter, we introduce the field of multi-robotics, providing a taxonomy for visualizing where contributions can be made in this field, and describing the essential domains of multi-robotics.

3 | Multi-robot systems



Introduction

In this Chapter, we dive into the field of robotics, and we discuss specifically multi-robot systems. As the term suggests, multi-robotics investigates the use of multiple robots typically performing distributed tasks or tasks that require the joint effort of multiple robots. In comparison to single robots, multi-robot systems provide higher scalability, flexibility, robustness and performance. Due to its enormous advantages, multi-robot systems have attracted over time the interest of an ever-increasing number of researchers, and it is still one of the most studied areas of robotics. Multi-robot systems can find application in both outdoor and indoor settings. In this dissertation, we analyze the problems when experimenting with a multi-robot system in indoor environments, and particularly in industrial environments. A taxonomy is presented to show the main topics of multi-robot systems and show in which of these domains the contributions of this thesis will be added. This Chapter represents a sub-activity of the DSRM, being the investigation of the theory.

This Chapter is structured as follows. In paragraph 3.1, we describe multi-robot systems and provide two classifications. In paragraph 3.2, we detail the main problems for the indoor application of a multi-robot system. In this paragraph, a taxonomy presents two approaches to deal with the identified problems, namely an *Analytical/Low-level approach* and a *Theoretical/High-level approach*. In sub-paragraph 3.2.1, the analytical approach is described, which consists of algorithms and design methods. In sub-paragraph 3.2.2, the theoretical approach is detailed, which includes micro- and macro-modelling. In paragraph 3.3, the main contributions of this master thesis are illustrated and shown in a comprehensive framework.

Together with Chapter 2, this Chapter builds the foundation on top of which the agent-based model is developed.

3.1 Classification of multi-robot systems

In the last decade, robotic technologies have been deployed for diverse employments, such as manufacturing, surveillance, exploration of human-dangerous environments, nursing and medical care, domestic support and entertainment. Recently, intelligent transport systems have also moved out from the testing ground and, in the near future, they will become commonplace. Among many robotic research areas explored over the years, the Multi-robot system (MRS) field has certainly received ample consideration. Although long established in the research, MRS is yet in the limelight of academics, as it represents the cross-point between two fields, Artificial Intelligence and Robotics, which have progressed exceptionally in recent years.

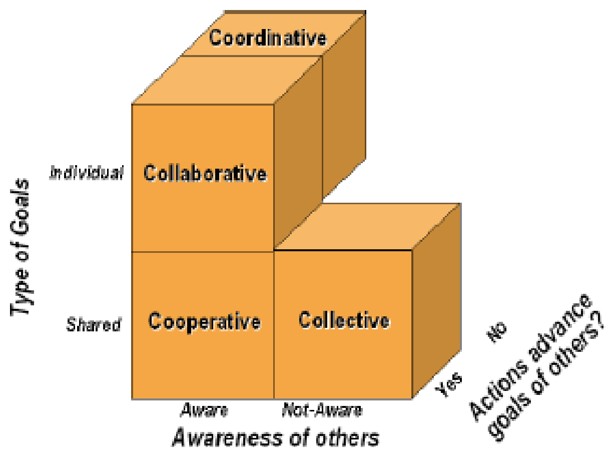


Figure 11. MRS classification, retrieved from L.E. Parker (2008)

MRS can be characterized as a field of research that investigates the use of multiple robots operating in the same environment. Robotic systems are mobile platforms, equipped with sensors and actuators, able to interact with other similar devices and with the environment in order to perform (simple or complex) tasks. Different classifications of MRS exist in the body of literature. However, the terms *Collective, Cooperative, Swarm, Collaborative* and *Coordinative Robotics* are still used interchangeably, although they present certain differences. In order to clear out the distinctive traits of these categories of MRS, we use the classification of L.E. Parker

(2008). The author divides the types of MRS along three different axes (Figure 11). On the horizontal axis, we view the awareness of others, which can be split into aware or unaware systems. In this context, awareness is not intended to whether robots are aware of the presence of their counterparts, like in L. Iocchi et al. (2000), but it refers to the ability of robots to discern the (future or present) actions and intentions of their peers. On the vertical axis, we find the type of goals, which can be either *individual* or *shared*. Along the diagonal axis, we find a yes-or-no question, which is whether the actions of a robot can help some of its teammates to achieve their goals or not.

Collective Robotics. In 1993, C.R. Kube and H. Zhang started working on a project, called the "Collective Robotic Intelligence Project", where the collective behaviors of social insects (e.g. bees, ants or cockroaches) were first studied and adapted to robots. According to the authors, the synergistic behavior of social insects allow to overcome the limited capabilities of unintelligent units. This works in accordance to the theories of "Swarm Intelligence" (G. Beni and J. Wang, 1993). In this type of systems, agents are not aware of the plans of other agents, but they share the same goals. Moreover, the actions of an agent produce utility for another agent. Swarm robotics belongs to this category of collective systems.

Swarm Robotics (SR). SR is a bio-inspired category of MRS that has become widely popular in recent years, to such an extent that it has ripened into a research field in its own right (M. Dorigo, 2014). SR is defined as "*the study of how large numbers of relatively simple physically embodied agents can be designed such that a desired collective behavior emerges from the local interactions among agents and between agents and the environment*" (E. Şahin, 2004). The concept of locality represents the core of the SR research. The local interactions between robots translate into global behaviors. This provides the robots with the ability to perform collectively tasks, which are too complex to be executed by single robots (I. Navarro and F. Matia, 2012).

Cooperative Robotics. Collective robotics, however, does not necessarily entail the cooperation among robots. There exist tasks which do not benefit from the cooperation of multiple agents, as a single robot is necessary and sufficient (G. Dudek, 1996). For example, tasks like localization and mapping do not require cooperation of robots. Cooperative robotics is a MRS where agents are aware of the actions of other agents, have the same common goals and the activity of an agent produces positive effects on the work of other agents. Cooperative robotics, thus, concentrates on the study of "robots that operate together to perform a given task" (A. Farinelli, L. Iocchi, D. Nardi, 2004). For instance, tasks like material-transport, box-pushing and pattern formation demand the cooperation of robots (M. Brambilla et al., 2013).

Collaborative Robotics. Collaborative robotics is closely related to cooperative robotics. However, the only distinction is that, in collaborative robotics, agents work together not to achieve a shared goal, but to better achieve the individual goals. An example can be that of heterogeneous robots with different capabilities and different goals, but that need to work together to optimize their own goals. However, from an observer that is not familiar with the capabilities of the robots, this can be viewed as a cooperative system.

Coordinative Robotics. Coordinative robotics refers to systems where agents are aware of the presence and actions of other agents, however they do not have a common goal and their work do not benefit from the activities of others. Coordinative systems are generally concerned with robots sharing the same workplaces, where robots need to minimize obstructions between each other.

MRS can be further categorized based on four main aspects: (i) the basic motivations for the use of MRS; (ii) hardware and software used; (iii) the tasks that robots should be able to perform; and (iv) the intended domains of application (A. Farinelli, L. Iocchi, D. Nardi, 2004). This classification can be further extended by including the population size, collective reconfigurability (i.e. adaptability of the system) and collective composition (i.e. homogenous or heterogenous agents) (G. Dudek, 1996). Using this classification, we can differentiate three types of multi-robotic approaches (Table 3), namely *cooperative approach*, *networked approach* and *swarm/collective approach* (S. Kernbach, 2013). The swarm approach has been earlier illustrated. The core of swarm robotics lies on the local knowledge and interactions of the robots, which increases the scalability, fault tolerance, flexibility of the system (E. Şahin, 2004; S. Kernbach, 2013, Brambilla et al., 2013). A cooperative approach also uses distributed sensing and considers hardware components less important. However, these systems may also have a centralized control, whereas a SR fully relies on decentralized control schemes. A networked approach is the one that presents the most differences from a SR. These systems heavily rely on high computational resources for sensing and communication. This allows robots to have always a global knowledge of the processes. In comparison to SR systems, other MRS are less minimalistic and put more emphasis on achieving high system performance. MRS have indeed originated from the motivation of improving the performance of a system through the synergistic behavior of their agents

(A. Farinelli, L. Iocchi & D. Nardi, 2004). Consequently, agents in other MRS can be equipped with more sophisticated sensor devices to improve the capabilities of robots.

Table 6. Differences Swarm, Cooperative, Networked robotics

System Dimensions	Swarm	Cooperative	Networked
Population size	N >> 1	N > 1 or N >> 1	N > 1 or N >> 1
Composition	Homogenous	Homogeneous or Heterogenous	Homogenous or Heterogenous
Hardware	Limited	Limited-to-Complex	Complex
Software control	Decentralized	Centralized-to-Decentralized	Centralized
Performance	Low	Medium-to-High	High
Scalability	High	Low-to-High	Low
Flexibility	High	Low-to-High	Low
Fault tolerance	High	Low-to-High	Low
Reconfigurability	High	Low-to-High	Low
Applications	Outdoor; unknown environments	Outdoor / Indoor; Known / Unknown environments	Indoor; Known environments

The scalability, flexibility, fault tolerance and reconfigurability (or adaptability) of cooperative systems depend on the team organization approach, which can be centralized or decentralized. When using a fully centralized approach, the scalability is reduced considering that all agents are connected to a central unit that becomes a bottleneck (constraint) to the system. The fault tolerance is lacking because if the central unit fails, the whole system fails (single-point-of-failure) (A. Khamis et al., 2015). Moreover, the flexibility and adaptability of MRS are also restricted, as fully centralized systems are typically optimized for a single configuration, and thus fail or obtain poor performance when changing scenario. On the contrary, decentralized approaches reduce the sensitivity of the system to the loss of a central unit, by transferring the control authority directly to robots. Moreover, as there is not a central unit, scalability, flexibility and adaptability are no longer an issue (A. Khamis et al., 2015, J.C. Barca and Y.A. Sekercioglu, 2013; E. Bonabeau and M. Dorigo, 1999). Nevertheless, under decentralized control, decisions are taken based on local communication and therefore a complete global knowledge is missing. Consequently, an optimal local solution may be equivalent to a sub-optimal global solution; therefore, decentralized approaches may produce sub-optimal outcomes.

Another main difference between these approaches lie on the domains of application, for which these systems are conceptualized. SR are developed for unknown environment and dangerous tasks. Examples where SR could be profitably deployed are mining, search and rescue, planetary and underwater exploration, and surveillance (M. Dorigo et al., 2014). Examples of swarm robot systems are shown in W. Liu et al. (2007) and A. Brutschy et al. (2014). In these systems, robots have limited local sensing and communication and cannot get global information of the environment. In these systems, robots perform actions based on environmental cues. For example, to transport objects, robots initially remain still with a probability *p* to switch to a random walk action. Robots walk randomly in the area, and as soon as they get in the proximity of an object, they switch behavior into move towards the object, then grab object and move to starting point. All actions are determined by external stimuli and robots keep moving randomly until they do not find something in the environment, not knowing the absolute position of objects (lack of global knowledge).

Conversely, other MRS could be additionally employed in indoor environments, for example for the transportation of objects or health care (E. Guizzo, 2008; R. D'Andrea, 2012; M. Shiomi et al., 2013). In Chapter 2, paragraph 2.6, we have described few implemented MRS employed in logistic environments. The Kiva system, the pioneer MRS in logistics, is an instance of a networked robotic system. The software control is executed via a multi-agent control system (P.R. Wurman et al., 2008), in which a job manager provides robots with instructions regarding the tasks to perform (*resource allocation*), while the driving units utilize learning algorithms to optimize their path and motion control. This system is not classifiable as cooperative, due to the low scalability and flexibility of the system that is only able to work in grid-like environments. The STO Express multi-robot parcel sorting system is another instance of a networked system, in which robots execute sorting tasks above an elevated grid. This system provides high performance but also poor flexibility, scalability and fault tolerance. The Hive, from Ocado, is another instance of a networked system. This system depends upon a centralized software control and robots have global knowledge of the environment. Furthermore, these robots have high sensing and computational resources (complex hardware) and improve their abilities through infra-robot information sharing.

J. Spletzer et al. (2001) develop the MARS (multiple autonomous robot system), which is an instance of a cooperative system, which relies on a hybrid control strategy. In this system, robots are employed for cooperative localization and object transportation tasks in indoor environments. This system exhibits average performance, high scalability, average fault tolerance (the system depends on robot leaders) and high flexibility and reconfigurability. Other examples of cooperative robot systems will be presented later in this Chapter.

In this research, we investigate the use of a cooperative multi-robot system, whose application domain is the transportation and sorting of parcels of different weight and size in a sorting hub (indoor environment). In the next paragraph, we describe the main problems when studying MRS in indoor environments.

3.2 Main problems for an indoor application of MRS

In the previous paragraph, we have provided a classification of different categories of MRS. Furthermore, we have disclosed the main benefits that MRS can provide in comparison to single robot systems. These benefits have captivated the attention of a myriad of researchers from diverse areas of study. Nevertheless, coordinating multiple robots sharing the same environment presents unique challenges that need to be solved before their deployment. The most challenging domains of MRS are notorious to researchers in the field (T. Arai et al., 2002; L.E. Parker, 2008; J.C. Barca and Y. A. Sekercioglu, 2013). These domains include:

- Communication and control schemes;
- Motion coordination (object transport, formation creation and retention);
- Localization and mapping;
- Path planning and obstacle avoidance;
- Task allocation.

Object transport and manipulation is typically a separate domain of research. Nevertheless, in this context, we are considering robots that do not need to manipulate objects, but that only transport objects from A to B. Therefore, the object transport becomes more of a motion coordination problem, while the object manipulation is completely absent. Therefore, we will analyze this problem as a motion coordination problem.

These research topics have all, to different degrees, been addressed over the years. Many authors have contributed in different ways to reduce the complexity of these problems. Generally, there are two opposite approaches to take when dealing with these topics (L. Bayindir, 2016), namely *Analytical / low-level approach* or/and

Theoretical / high-level approach (Figure 12). The Analytical / low-level approach is concerned with the development of algorithms for specific desired goal-directed behaviors of robots, or with the design of the architectures for MRS. Opposite, the Theoretical / high-level approach serves to describe the system operations at a more abstract level. In MRS, this typically matches with the modeling and simulation of a robotic system. Modeling can be further categorized in micro- and macro- modeling. The micro-level concerns with the study of the individual behavior of interacting agents, while the collective properties of the macro process are neglected. The macro-level, instead, concerns more with the analysis of the wider, system behavior than with the properties of individual robots. Thus, by using macro modeling, the collective behavior of the entire system becomes visible to an external viewer (M. Brambilla et al., 2013).

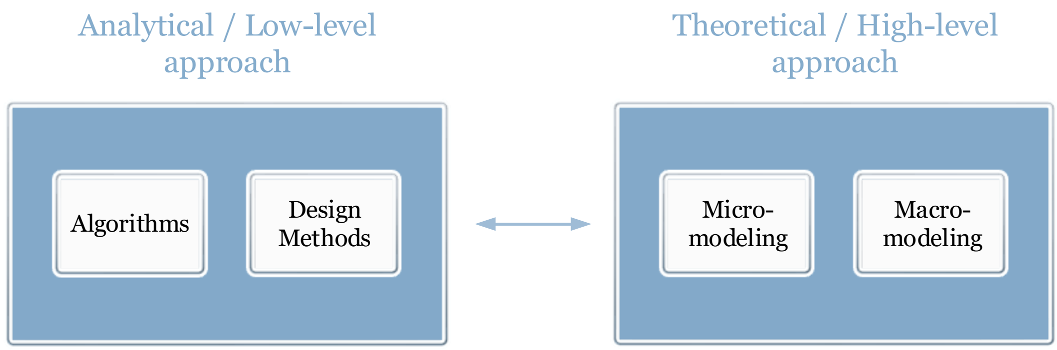


Figure 12: Taxonomy for MRS study approaches

The goal is to classify the articles published on the above-mentioned topics related to MRS according to the different approaches used to design or analyze these systems. The analysis of the existing field of knowledge form the foundation of this dissertation, and the thesis contributions will help cover some of the gaps found.

3.2.1 Analytical / low-level approach

The Analytical / low-level approach is concerned with the development of mathematical algorithms for specific desired behaviors of robots (*software*), or with the design of new architecture methods (*software and hardware*). Design methods are often used analogously with algorithms. In this case, we consider design methods as the phase where software and, mainly, hardware solutions are proposed to meet specific system requirements. Thus, a clear distinction between algorithms and design methods exist in this dissertation.

3.2.1.1 Algorithms

The algorithm development phase is seemingly the most challenging and the most examined in the literature, as proven from the extensive study research carried out on the topic. Algorithms can vary from task to task, while certain algorithms can be employed for multiple tasks. We will now shortly analyze a limited number of algorithms used in the literature based on the tasks they are designed to tackle (Table 4).

Table 7. Algorithms for each MRS domain

MRS Domains	Algorithms
Motion coordination	Leader-follower; Behavioral; Virtual structure; Graph-based; Artificial potential field
Localization and mapping	Voronoi partitioning; Cortés's Coverage coordination; SLAM
Path planning and collision avoidance	Breadth First Search; Greedy Best First Search; Dijkstra; A*; DPC; CDP; DCOP; ADOPT; M*; Basic Theta*; Phi*
Task allocation	Hungarian Method; OAP; AEP; market- / auction-based algorithms; SPP; min-max heuristics; ASyMTRe; ASyMTRe-D

Motion coordination. Y. Zhang and H. Mehrjerdi (2013) provide a survey on control and coordination of multiple unmanned vehicles in normal and fault situations. The authors describe different strategies that have been developed over the years for the coordination of multiple vehicles using group behaviors. These coordination strategies include the *leader-follower* approach, the *behavioral* approach, the *virtual structure* approach, the *graph-based* approach and the *artificial potential field* approach.

The leader-follower approach has been extensively used for the control and coordination of multiple mobile robots (J.P. Desai et al., 1998 and 2001; H. Sira-Ramirez and R. Castro-Linares, 2010; N. Noguchi et al. 2004; C. Zhang et al., 2016). Using this method, one robot is assigned the role of leader, while the rest are followers. The followers need to position at a relative distance with respect to the leader. H. Sira-Ramirez and R. Castro-Linares (2010) present a formation control algorithm, which enables the leader robot to follow a given path and a follower robot to track the leader's path, follow it while keeping a certain separation from the leader. C. Zhang et al. (2016) use a leader-follower approach to improve the efficiency of two robots in an agricultural fieldwork. In comparison, J.P. Desai et al. (1998) show a system with three mobile robots, with the third robot maintaining the desired distance from two leaders. M.A. Kamel and Y. Zhang (2015) illustrate two techniques called separation-separation and separation-and-bearing to improve the stability of the formation. Separation-separation concerns systems where the follower needs to maintain a given distance from the leader. Separation-and-bearing entails that the follower has to maintain a given distance and angle from the leader. The advantage of this approach is that it is simple, reliable and scalable (Y. Zhang and H. Mehrjerdi, 2013; C. Zhang et al., 2016; L. Consolini et al., 2008). The disadvantage is its low fault-tolerance, considering its centralized nature.

The artificial potential approach derives from the flocking and schooling techniques used by birds and fishes to move in aggregates. In this case, each robot exerts virtual attractive and repulsive forces on other neighboring robots that allow them to constantly place near and move away from a virtual leader (N.E. Leonard et al., 2001). Birds and fishes exhibit cooperative motion. They aggregate in groups and move together by applying three steering actions (Z. Qu, 2009). First, they move toward the center of mass of the closest neighborhood (cohesion). Second, they move away from the closest entity in order to avoid collision (separation). Third, they adjust their heading and speed according to the average heading and speed of the other agents (alignment). The three rules are used to obtain collision avoidance, velocity matching and flock centering (Brambilla et al., 2013). This enables robots to move in a synchronous way. The advantage of this approach is that it is fully decentralized, in thus it has high tolerance to failure and is highly scalable. By contrast, this algorithm does not allow to keep fixed distances between robots and to follow a given path in a timely way. Furthermore, the predictability of the system is also low, meaning that the robots change relentless their states, behaviors, or the like, making it difficult to know in advance what to expect from the systems.

The graph approach refers to the study of mathematical structures called graphs, i.e. a collection of nodes and a collection of arcs that connect the various nodes. Arcs can be either directed, where the flow through the arc is allowed in only one direction, or undirected arcs, where the flow through the arc is allowed in both directions (C.T. Ragsdale, 2011). Graph theory has been applied for the coordination of multiple robots, also together with follower-leader approach. J.P. Desai et al. (1998) use graph theory for the formation retention between leader and followers, showing three different graphs of formation. The authors argue that in the presence of obstacles, it may be necessary to switch from one formation to another, and this leads to non-isomorphic formations between leaders and followers. A. Jadbababie et al (2003) combine graph theory and dynamical system theory to describe a continuous-time leader-following approach. The authors focus on the study of T. Vicsek (1995) on flocking of birds and apply to it graph theory. R. Olfati-Saber et al. (2004) make a similar study about networks of dynamic agents with continuously switching formations.

In the virtual structure approach, the formation is considered as a single entity. A desired collective motion is provided to the structure via a centralized system. J. Ghommam et al. (2010) present a formation path following algorithm using a virtual structure approach. In this case, the formation of the robots is treated as a single rigid body that moves into desired configuration. Robots use feedbacks to prevent a member to leave the formation. The advantage of this method is that it is less dependent on one agent. The main disadvantages are its centralized nature, large inter-robot communication bandwidth required, and low scalability (L. Consolini et al., 2008; I. Mas and C. Kitts, 2010). I. Mas and C. Kitts (2010) prove that introducing additional robots to formations, using a virtual structure approach, affects the structure / physics of the rigid body. Therefore, the authors concluded that virtual structure algorithms offer lower scalability compared to leader-follower algorithms.

The behavioral approach is based on the stigmercy theory (P.P. Grassé, 1959), where the coordination of robots is achieved through the local perception and indirect communication of agents (C.R. Kube and E. Bonabeau, 2000). The main disadvantage of this approach is that it is mathematically hard to guarantee the stability of the group formation (J. Ghommam et al., 2010).

Localization and mapping. Localization and mapping deal with the problem of obtaining spatial models of physical environments through MRS. Using teams of robots for creating maps of a place is renowned as one the biggest advantages of MRS in comparison to single-robot systems (S. Thrun, 2002). The localization and mapping problem can be tackled by using a coverage coordination algorithm, where the environment is partitioned in non-overlapping areas based on the initial positions of robots. Voronoi partitioning has been used in many articles (M. Schwager et al., 2009; A. Breitenmoser et al., 2010). J. Cortés et al. (2004) propose a multi-robot coverage technique where robots communicate their positions with neighboring robots to guarantee complete and non-overlapping partitioning of the environment, based on Lloyd’s algorithm. K. Hungerford et al. (2016) propose a novel algorithm that enables robots to cover inaccessible portions of Voronoi cells due to obstacles through the coordination of robots actions and a repartitioning of the space. A. Howard (2006) develops a simultaneous localization and mapping (SLAM) algorithm for multi-robot systems, which was previously applied only on single-robot systems. This cooperative multi-robot mapping technique assumes that when robots are far from sensor-sight, they continue mapping the area with individual exploration strategies. However, when robots are within sensor range, they are able to detect one another and exclude the observations that are achieved by other robots. Thus, robots broadcast their observations to one another, and are able to mutually recognize each other and their respective positions.

Path planning and obstacle avoidance. In MRS, path-planning and collision avoidance are central problems that need to be considered as one (B.H. Lee & C.S.G. Lee, 1987). The multi-robot path planning algorithms aim at determining the path that each robot should take to reach its goal, while avoiding collisions with other robots and obstacles. These algorithms correspond to optimization methods that strive to minimize the total path length, the total time or the energy to reach goals. There are several algorithms that can be used to solve the path-planning problem, the most popular being the graph search methods such as the Breadth First Search, Greedy Best First Search, Dijkstra and A*. S. Bhattacharya et al. (2010) present a new algorithm called DPC (Distributed Path Consensus), which calculates an efficient way of finding optimal paths for multiple heterogeneous robots under time-parametrized distance constraints. The DCP algorithm implies running a series of graph searches and, at each run, the search finds a path that minimizes the weighted sum of the path costs and the amount to which the identified path violates the constraints for the paths of other robots. Another algorithm called CDP (Cooperative Distributed Planning) is used for path planning where multiple agents communicate and coordinate their actions with the objective to reach a consensus on paths. This resembles the market- or auction- based algorithms of task assignment, but applied to path planning. DCOP (M. Yokoo and K. Hirayama, 2000) and ADOPT (S. Bhattacharya et al., 2010) are two other commonly used algorithms for multi-robot path planning. In DCOP, each path for an agent is considered as its state. In this case, this algorithm is computationally highly expensive. The ADOPT algorithm uses sequential searches and asynchronous communication between agents until a path is attained (S. Bhattacharya et al., 2010). G. Wagner and H. Choset (2011) address the multi-robot path planning problem using a graph search algorithm called M*. With this algorithm, initially each robot identifies its individually optimal path. When robot-to-robot collisions are found, the robot enlarges its search space until a collision free path is found.

Task allocation. Multi-robot task allocation is one of the most challenging and most investigated domains of MRS, which deals with the way robots are assigned to the tasks in such a way that the system performance is optimized and constraints are satisfied. B.P. Gerkey and M.J. Mataric (2004) propose a taxonomy that is useful in order to distinguish the type of algorithms that can be used for different categories of MRS. They describe multi-robot task allocation problems based on three determinants: single-task (ST) vs multi-task (MT); single-robot (SR) vs multi-robot (MR); and instantaneous assignment (IA) vs time-extended assignment (TA). In this case, we only consider problems with instantaneous assignment, since time-extended assignment problems are more related to scheduling problems than assignment problems. ST-SR-IA is the most well-known and simplest problem, where: each robot is able to perform only a single task at a time; each task only requires one robot to be accomplished; and the allocation of the tasks to the robots is instantaneous, meaning that the information is not good enough to plan future allocations. Generally, these problems can be simply solved by using mixed integer programming algorithms. The most basic versions of MIP algorithm is the Hungarian Method (H.W. Kuhn, 1955) or Optimal Assignment Problem (D. Gale, 1960), where a one-to-one assignment is made in such a way that the overall objective function (e.g. cost, distance, profit etc.) is maximized or minimized. A variant algorithm used for these problems is the ALLIANCE Efficiency Problem (AEP), where multiple robots with different skills try to obtain essential resources from an environment to survive (V. Lattarulo and G.T. Parks, 2012). These optimization-based algorithms are used in centralized systems, where the resources are typically known in advance. These problems can also be solved using (partially or fully) decentralized algorithms. The two most famous approaches used in this case are the market- or auction-based algorithms. These algorithms involve a negotiation process between robots based on market theory (R.M. Zlot, 2006; L. Luo et al., 2015). Robots bid for tasks based on their capabilities, and based on their bids and the auction, tasks are assigned to the set of robots (A. Khamis et al., 2015).

ST-MR-IA considers problems where each robot is able to perform only a single task at a time (ST), but each task requires the combined effort of multiple robots. These types of problem are largely more difficult than the previous problems (B.P. Gerkey and M.J. Mataric, 2004). These problems present very complex task decomposition (NP-hard) and only few researchers have attempted to solve these task assignment problems. The most employed algorithm to cope with them is the Set Partitioning Problem, which divides a set of robots in finite sets of feasible teams, each of which try to optimize its own utility. However, as underlined in O. Sheohory and S. Kraus

(1995), the SPP solutions have two main deficiencies, namely (1) the computation complexity is exponential in the number of robots, (2) these solutions are centralized, i.e. formation of coalitions can be calculated and implemented only by a central computer. To address these problems, the authors propose a distributed set-partitioning algorithm, with agents calculating and forming coalitions without the assistance of a central agent. Indeed, agents create coalitions based on their capabilities and the requirements of the tasks. The main disadvantage of this algorithm is that the average computational complexity is high and the solution does not scale well with increasing number of agents. However, this solution decreases the inter-agent communication. This algorithm can be used also for combinations of ST-SR-IA and ST-MR-IA tasks. L.E. Parker and F. Tang (2006) develop an algorithm, termed ASyMTRe (Automated Synthesis of Multirobot Task solutions through software Reconfiguration), to address the ST-MR-IA multi-robot task allocation problem. The objective of this algorithm is to solve ST-MR problems by forming coalitions, i.e. by organizing multiple robots into subgroups to accomplish a given task. This centralized algorithm is used for solving tasks that cannot be handled by single robots. The collaboration among robots is achieved by using robot schemas. Each robot contains a schema, i.e. a control framework that includes inputs, outputs, local variables, behaviors. The robot schema defines how the input needs to be processed in order to generate a certain output. The ASyMTRe algorithm builds a network of schemas connecting the outputs of one robot schema to the inputs of other robot schemas. In this way, computations from multiple schemas of robots are summed up and normalized to produce the desired collaborative behaviors. Furthermore, the authors develop a distributed version of this algorithm, termed ASyMTRe-D, which produces more reliable and flexible results, but it lacks of quality solutions. However, heterogeneous robots are used to perform tasks and coalitions are formed according to the capabilities of robots. Moreover, tasks are assigned sequentially in the experiments. Therefore, at time 1, task 1 is auctioneed, while at time 2 and 3, task 2 and 3 are auctioneed respectively. When the coalitions for these tasks are determined, other tasks are announced. This implies a considerable number of idleness periods for robots. Furthermore, considering that some robots are more capable than others, when these robots are already performing other tasks, the less capable robots need to wait until the accomplishment of said tasks to form coalitions with capable robots. J. Guerrero and G. Oliver (2012) propose another solution to address ST-MR-IA tasks. This task assignment problem is addressed using an auction-based algorithm in which the robot that discovers first the tasks becomes the leader and holds an auction to find other robots. In this algorithm, every task has a single leader that calls an auction in which the other members of the coalitions are decided based on their work capacities. When the leader decides which robots can be part of the coalition, it sends a confirmation to these robots and wait until it receives a response. The leader also decides the adequate group size for the execution of a task. This algorithm is able to decrease the computational and communication complexity of the task to address. However, in this research, the authors only focus on finding a solution for ST-MR-IA tasks.

Therefore, the performance of ST-SR-IA and ST-MR-IA tasks within the same domain of application, using homogenous robots and without sequential task assignment represents the knowledge gap for this thesis. It is important to notice that the other types of problems where robots can perform more tasks at a time (time-extended assignments) are neglected in this thesis.

Summary. It is apparent that a broad array of algorithms exist in the literature, with several algorithms being leveraged to address the domains of MRS. Pertaining to motion coordination problems, we can conclude that the leader-follower, virtual-structure, and graph-based structure (which can be used in combination with the other two) algorithms are suitable for an indoor application of MRS.
These algorithms lack fault-tolerance given their centralized nature, but guarantee the stability of the collective motion. Voronoi and Coverage Coordination algorithms can be used to address localization and mapping problems. These algorithms ensure that the coordination and communication of robots is distributed. The distributed sensing capabilities of the robots promotes parallelism, and communication between robots accelerates the mapping process. Path planning and collision avoidance in MRS are studied side by side considering that a poor planning of path can have adverse consequences on collision avoidance, and vice versa. A large array of algorithms belong to this domain of MRS, with the most popular being heuristics and A* pathfinding algorithms. These algorithms are mainly used for single robots applications and with static obstacles. Nevertheless, they have also found application in MRS in cooperation with collision avoidance techniques. Task allocation is another crucial sphere of MRS and can be categorized using three determinants: single-task (ST) vs multi-task (MT); single-robot (SR) vs multi-robot (MR); and instantaneous assignment (IA) vs time-extended assignment (TA). Considering the scope of our research, we focus on ST-SR-IA and ST-MR-IA. The former task assignments problems are not NP-hard and can be simply solved using optimization algorithms such as Hungarian or Optimal Assignment Problem (OAP), or market- or auction-based algorithms. The

latter are NP-hard that require complex algorithms, such as ASyMTRe or ASyMTRe-D, which act at control level to assign tasks to multiple robots. A solution to combination of ST-MR-IA and ST-SR-IA problems, however, represents the main knowledge gap for this thesis.

3.2.1.2 Architecture and Control Design Methods

In this paragraph, we focus on the architecture and control design methods for MRS. These design methods aim to provide the single robots with the computing, sensing and communication capabilities they need in order to collect data and make real-time decisions. Furthermore, the individual capabilities of robots are enhanced by sharing perception, computation and actuation capabilities with each other.

Control Architecture. In multi-robotics, typically three control architectures are singled out: *reactive*, *deliberative* and *hybrid* control systems (S. Kernbach, 2013).

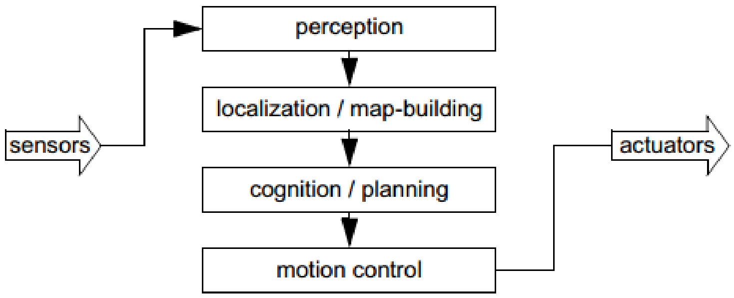


Figure 13: Model-based architecture, retrieved from R. Siegwart et al. (2011)

Deliberative control architectures follow the “*think hard, then act*” or “*sense-plan-act*” method. The deliberative control architecture is commonly referred as “*model-based*” control approach given its strong dependence on accurate and complete world models. Robots with this architecture exploit maximally their sensory information and internal states to create a plan of action. Therefore, these robots have a high predictive capability to look ahead and solve the “what if - then” paradigms by selecting the most befitting behaviors for the tasks at hand (S. Verret, 2005; R.C. Arkin, 1998). Figure 13 features an example of model-based or deliberative control architecture (R. Siegwart et al., 2011).

Reactive control architectures follow the “*do not think, (re-)act*” method. This is also referred as “*behavior-based*” control approach, given its nature inspiration. In reactive control, sensors and actuators are tightly coupled to provide the robots with the ability to re-act rapidly to changing environmental conditions (“stimulus-response”). This control approach is used in *swarm robotics*, where robots have low-level intelligence, with no learning ability and no internal representation of the environment in which they are placed (R.C. Arkin, 1998; E. Şahin, 2004). Reactive robots instantaneously convert sensory inputs into motion vectors in order to avoid obstacles, move towards a goal and maintain a formation. Figure 14 features a behavior-based control architecture (R. Siegwart et al., 2011)

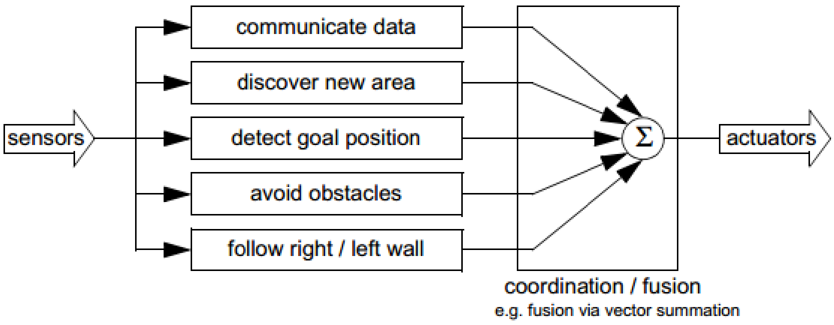


Figure 14: Behavior-based architecture, retrieved from R. Siegwart et al. (2011)

Hybrid control architectures are a mixture of the two and they aim to enhance the capability of robots by combining the advantages provided by deliberative control systems with those given by reactive control systems. This integration of the two methods gives robots the ability to “*think and act in parallel*” (M.K. Sahota et al., 1994). Thus, hybrid architectures integrate the responsiveness, robustness and flexibility of purely reactive systems with the performance of purely deliberative methods.

R.J. Firby (1990) developed a three-layer robot architecture to combine reactive and deliberative control methods. At low level (reactive layer), sensors and actuators are tightly coupled to give robots the ability to react rapidly to changes in the environment and execute reactively actions such as obstacle avoidance. At intermediate level, robots accomplish long-term tasks, such as path planning or localization and mapping. At higher level, robots coordinate their actions to perform asynchronous tasks like cooperative search or synchronous tasks like cooperative transport (S. Verret, 2005). This control architecture has been used in A. Marino et al. (2013) for multi-robot patrolling applications.

In the literature, there are many other examples of control architectures for MRS, such as ALLIANCE (L.E. Parker, 1998), CHARON (R. Fierro et al., 2002), 3T (D. Schreckenghost et al., 1998), CLARAty (Volpe et al., 2001), CAMPOUT (T. Huntsberger et al., 2003), CEBOT (A.H. Cai, 1995) and others (B. Hichri et al., 2016). These architectures have focused on providing robots with cooperative behaviors.

The choice of the architecture and control method strongly affects the performance of the overall system. The system performance increases with the degree of deliberation and coordination embedded in the robot system. High level of deliberation and coordination require substantial sensing and communication capabilities.

We now examine the state-of-the-art in sensing and communication capabilities (hardware components).

Communication technologies. Robots can communicate with each other within a certain distance. The communication range is generally conceptualized as a circle with a certain radius, which indicates the power signal, originated from the emitter. The communication signal attenuates exponentially with distance, following the equation of Y. Chen and H. Kobayashi (2002). Communication in MRS can occur via radio, light or sound. In MRS, obtaining a reliable robot-to-robot communication is a major challenge, considering that many mobile devices and fixed devices communicate on the same channel.

Robots mainly communicate with their teammates using radio communication, e.g. wireless LAN/DSRC (Dedicated Short Range Communication) technology based on the IEEE 802.11 protocol (F. Cali et al., 2000). Wireless communication technology provides automatic establishment of ad hoc robot-to-robot network within a range of 300 m with a very small latency of 0.2 s. This small latency is especially suitable in safety critical applications. Furthermore, the high data rate of 20 Mb/s enables a robot to rapidly exchange data with other autonomous agents. Low-power alternatives to wireless technology are Bluetooth and ZigBee (S. Kernbach, 2013). Bluetooth devices can provide communication ranges from 1 to 100 m and data rates from hundred Kb/s to a maximum of 10 Mb/s. ZigBee devices can operate at up to 250 Kb/s in the 2.4 GHz band, while data rate ranges from 10 to 75 meters. The downside of the Bluetooth technology is the limitation to communicate with only seven devices at a time, while ZigBee was used in an experiment with a swarm of UAVs (B.J. Julian, 2009). P. Ivanov and A. Shell (2016) explain how robots in a multi-robot system can communicate using two-channel systems (e.g. wireless and infrared). Both channels can transmit a robot's ID to another robot in a reliable way. In addition, the infrared channel is used to detect distance and position to other robots.

Communication devices are crucial for localization, i.e. to determine the position of robots in an environment. In outdoor settings, a robot's absolute position can be obtained using GPS, which gives a horizontal precision of few meters and vertical precision of tens of meters (J.C. Zufferey et al., 2013). The localization precision can be enhanced using other techniques such as WAAS (Wide-Area Augmentation System) or DGPS (Differential GPS). DGPS provides the system with an additional signal from a ground based system in conjunction with the conventional GPS signal, to more accurately determine the location and position of an autonomous robot and keep track of its movements (Z. Yang et al., 2000). However, in indoor settings, GPS or the other mentioned techniques cannot be leveraged due to weak signals and multi-path reflections (R. Siegwart et al., 2011). Absolute positioning indoor can be implemented using vision cameras or infrared (J. Oyekan and H. Huosheng, 2009). This technology requires the installation of static reference points, i.e. beacons. Another novel approach to localization in indoor environment is termed SLAM (Simultaneous Localization and Mapping), which method is used to simultaneously map an unknown environment and keep track of the position of the robot. This technique does not require the installation of static indoor devices like beacons (H. Durrant-Whyte and T. Bailey, 2006). Presently, this technique has been experimented only on small groups of robots or on single robots (S. Kernbach, 2013).

As already stated, these communication approaches are used to acquire the absolute position of a robot in the reference frame. Nevertheless, localization implies not only knowing the absolute position of a robot in the space, but also its relative position with respect to other objects or targets in the environment. To do so, robots need to use their sensing technologies and compute their relative positions to the objects.

We now describe the sensing technologies used by robots to measure their relative distance to other objects.

Sensing technologies. A sensor is a technological instrument used to measure the presence and characteristics of objects, and to calibrate the internal values of a robot. Multiple sensors need to be equipped on-robot in order to see and interpret the surrounding in the same way, as a human being is able to do.

Sensors can be categorized based on whether they are proprioceptive or exteroceptive (A. Discant et al., 2007). Proprioceptive sensors measure internal values of a robot, such as speed, energy, steering angle and so on. By contrast, exteroceptive sensors are used to get data regarding the environment, such as presence of or distance from other objects, light, sound and so on. In addition, sensors can be further categorized in passive or active. Passive sensors only acquire data from the environment (e.g. cameras, microphones). Active sensors perform actions, for instance they emit radio waves to check the presence of other objects, and acquire data regarding the environmental reactions, e.g. measure the distance from an object.

Table 8. Sensors characteristics

Sensor characteristics	Description
Range	Combination of distance [meters] and horizontal and vertical detection angle [degrees]
Operating conditions	Conditions under which the sensor will or will not be able to function, e.g. temperature, humidity, precipitation, dust, etcetera
Detection rate / Accuracy	Percentage of correct detection of an object
False alarm rate	Percentage of detections that does not correspond to the object
Resolution	Capability of the system to distinguish object characteristics
Numerical error	Error to distinguish object characteristics, given in terms of confidence interval
Bandwidth or frequency	Number (or speed) of measurements per seconds
Cost	Economic value of a sensor

R. Siegwart et al. (2011) provide a classification of the most used sensors for robot applications. This classification includes tactile sensors (e.g. contact switches, bumpers), wheel/motor sensors (e.g. rotary encoders), heading sensors (e.g. gyroscopes), ground-based beacons (e.g. ultrasonic beacons), active ranging (e.g. short/long-range radars, LIDAR), motion/speed sensors (e.g. Doppler radar), vision-based sensors (P. Hintenaus, 2015; R.H. Rasshofer and K. Gresser, 2005).

The sensor characteristics and requirements greatly vary according to the operations the robots are called to handle (Table 5). Generally speaking, the sensor characteristics that are used as guideline to formulate sensor requirements are the following: range, operating conditions, detection rate, false alarm rate, resolution, numerical error, bandwidth/frequency and cost (A. Amditis et al., 2012).

Summary. In this paragraph, we have illustrated the main control, communication and sensing design methods utilized in MRS. Three control architecture approaches have been pointed out, namely deliberative, reactive and hybrid. Deliberative control architectures exploits maximally sensory information to create plans of action. Therefore, robots designed with these architectures are able to sense the environmental inputs, plan their actions and then select the most appropriate behaviors according to the action to handle. Reactive control architectures provide robots with the ability to convert rapidly sensory inputs into motion vectors in order to re-act to changing environmental conditions. The performance of a system increases with the increase of deliberation of robots, whereas robustness, flexibility and responsiveness increases with the increase of reactivity of robots. Hybrid control architectures combine the strengths of deliberative control with those of reactive control architectures to provide MRS with a good balance of performance, robustness, flexibility and reactivity.

3.2.2 Theoretical / high-level approach

Modeling and simulation are among the most used tools to analyze and validate the results of MRS. Micro-modeling focuses on each robot individually and studies robots' interactions with other robots and the environment. Macro-modeling, instead, does not take into account the individual behaviors of the robots, but considers the system as a whole. Thus, macro-modeling concerns the description of the MRS at a higher level, detaching the view from the small elements of the system in favor of a major consideration of the bigger picture.

We now present some works found in the literature where micro- and macro- models of MRS are analyzed.

3.2.2.1 Micro-modeling

Micro-models are developed to test the accuracy of algorithms and design methods (Paragraph 3.3.1). Micro-models can focus on a specific task, or behavior, or simply on the communication and control architecture of robots. Here, we analyze only few examples of micro-models developed in the literature based on the tasks they are designed to address. Many other noteworthy studies can be found in the literature, but they are not described in this dissertation due to time constraints.

Motion coordination. J.P. Desai et al. (2001) address the issue of controlling and coordinating a team of non-holonomic mobile robots. The authors modelled the behaviors of individual robots to allow them to move in a space retaining a desired formation and changing it when required, using graph theory (see Figure 15). They demonstrated the validity of the developed control laws by showing how the transition from one formation to another occurs in the presence of obstacles. In order to do so, they decompose the problem into two sub problems, namely (1) controlling a leader robot and (2) controlling other follower robots in the teams. The results show how a team of three and six robots can change their formation when finding obstacles.

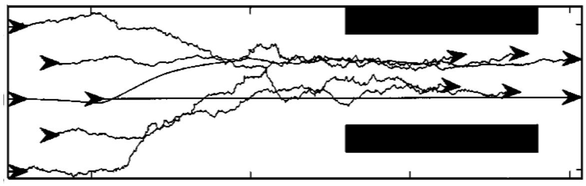


Figure 15: simulation of formation pattern retention method, retrieved from J.P. Desai et al. (2001)

C. Zhang et al. (2016) develop a micro-model to show the validity of a designed leader-follower algorithm for two robot tractors that perform agricultural fieldwork. The results of the simulation show that it is feasible to use two robot tractors to work coordinately in one field, without collisions. Furthermore, by installing a laser scanner and a bumper switch on-robots, they show that collision is avoided also when robots meet other non-autonomous objects (e.g. humans). This simulation experiment also provides useful information regarding the work efficiency, i.e. reduction of total work time, of the leader-follower algorithm.

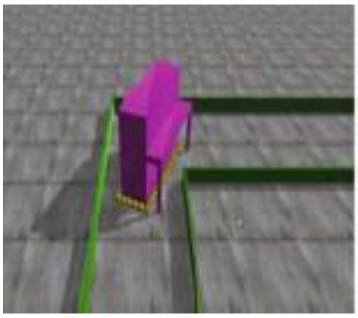


Figure 16: Cooperative transport of heavy objects, retrieved from Z. Wang and M. Schwager (2016)

Z. Wang and M. Schwager (2016) propose a decentralized algorithm for a multi-robot transportation problem, involving a large number of small robots moving a large object along a desired path until reaching the final destination. Two simulation experiments are implemented, one with twelve robots transporting a long rectangular plank and one with one thousand robots transporting a piano (Figure 16). The goal of the simulation is to show how to coordinate the actions of multiple robots to transport heavy objects along a desired trajectory. The number of robots is selected based on the sum of all forces of robots necessary to transport the object until destination. In the examined situations, robots only know parameters regarding their own personal features, but do not have global knowledge about the environment or the presence of other robots.

Moreover, the leader robot only knows the desired trajectory and the coordination occurs via a consensus-achievement strategy.

B. Hichri et al. (2016) also address the problem of cooperative transportation of heavy or differently shaped payloads by a fleet of homogenous mobile robots. The minimum number of robots required to lift and transport an object is again obtained by measuring the sum of all vertical tangential lifting forces of the robots that needs to be equal to the mass of the payload multiplied by its gravity force. In this case, the authors use simulation to ensure that robots are able to transport the payload until a final destination and that the payload lifting occurs without losing the stability of the formation.

Localization and mapping. K. Hungerford et al. (2016) demonstrate the accuracy of the developed algorithm for the problem of coverage path planning in an initially unknown environment, simulating robots within the Webots simulator. In this simulation, robots use proximity sensors (short-range radars) to avoid obstacles, Bluetooth protocol for robot-to-robot communication and GPS for localization. Four different scenarios, with changing environments and number of robots (5-7 robots), have been used to prove the success of the algorithm to allow the robots completing the coverage of the entire environment.

P. Koch et al. (2016) applies a 2D SLAM in combination with a 2D LIDAR to a MRS. This integration of technologies provides the system with the ability to receive extremely accurate details regarding the position of robots and increase the awareness of the environment and of dynamic objects. Using this approach, multi-robots are shown able to build a joint map in parallel instead of merging occasionally smaller maps. In order to test the feasibility of this novel approach, the authors implement the multi-SLAM framework in the ROS Simple Two Dimensional Robot Simulator (STDR). Four robots are simulated in an indoor environment, which start at the same time and construct the map of the environment in parallel. The simulation experiments demonstrate the high performance of the MRS to build a complete map of the environment using a multi-SLAM framework in combination to LIDAR technology.

D.L. Martínez and A. Halme (2016) develop a NetLogo model to tackle the simultaneous tasks of creating a map and recharging robots' power units. This simulation consists of a fleet of MarsuBots exploring the space and recharging themselves with the aid of a MotherBot. The MotherBot is depicted as a tank-like robot that is capable of hosting up to three MarsuBots and charging them. In the developed simulation scenario, eight MarsuBots scanning the surroundings using a laser rangefinder, marking the explored space, and avoiding static objects (Figure 17). When the battery level of a robot drops below a predefined threshold, a robot goes back to the MotherBot, positions itself in a queue if required and waits until the MotherBot recharges its battery. After recharging, this robot continues mapping the space using the same exploration actions.

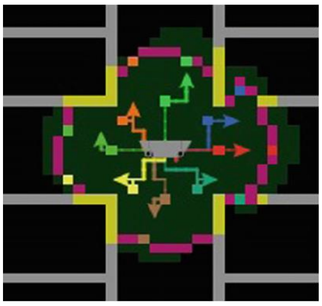


Figure 17: NetLogo simulation of MarsuBots, retrieved from D.L. Martínez and A. Halme (2016)

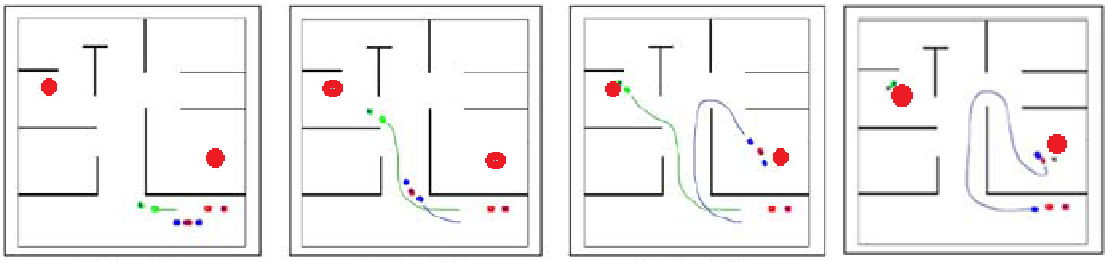
Path planning and collision avoidance. F. Duchon et al. (2014) address the path planning problem for a mobile robot based on a grid map. A* is one of the most known path planning algorithms in robotics and it is typically used for a grid configuration space. This algorithm allows finding the optimal path based on connection between the closest cells. Nevertheless, the basic A* does not allow searching in every angle. In order to do so, it must be extended with other algorithms (e.g. Basic Theta* and Phi*). In this paper, the authors test the extended version of A* to the path finding problem in three simulated scenarios. The results show that this algorithm is able to find the shortest paths but it requires more computational time in comparison to the basic A*.

R. Regele and P. Levi (2006) assess the fitness of their distributed path planning approach using simulation. The path planning algorithm used is based on cooperative negotiations between robots. The idea is that of a cooperation strategy among robots, where each agent does not act in a selfish way, but communicates its intentions with its conflicting peers in order to come up with a collision-free path solution. Furthermore, the developed approach also takes into account the diverse priorities (urgencies) of the robots. For example, when a robot is informed that another robot shares the same path, but the latter has higher priority, the former will let him pass first. The authors simulate the MRS under several scenarios to test the performance and scalability of the algorithm. The results show that the proposed algorithm is efficient, reliable and scalable.

C. Cai et al. (2007) use simulation to demonstrate the effectiveness of the developed path planning and collision avoidance (predictive-) algorithm, based on omni-directional vision techniques that allow robots to identify static obstacles and velocity of other robots. The simulation environment analyzed in this research consists of five robots, ten static obstacles and five goals assigned to each robot. Within this simulation environment, robots need to avoid collisions with static objects as well as avoid collisions with other moving robots. By assigning different priorities to the robots, robots predict their safe trajectories, avoiding successfully all the obstacles in the system.

Task allocation. L. Lin and Z. Zheng (2005) present a combinatorial bids based MRTA algorithm that allows robots to form groups and build cooperative bids to fulfill complex tasks. The combinatorial bid process can be divided into six main processes, namely task announcement, bids submission, task pre-award, bids combination, com-bids confirmation and task allocation. To test the feasibility of this method, the authors implement a simulated scenario, which consists of a meeting room where many desks and chairs are randomly distributed in the space. The task is to put the desks and chairs to specific places in order. The analysis of the collected simulation results show that the combinatorial bids method outperforms the typical auction-based methods for both tasks that require the cooperation of multiple robots and for tasks that do not require the cooperation of robots.

L.E. Parker and F. Tang (2006) validate their ASyMTRe and ASyMTRe-D algorithms by simulating a complex multi-robot task allocation mission (Figures 18a-18b-18c-18d). In this simulated environment, robots are initially placed at a starting position and they need to move towards the tasks to accomplish (red dots). All robots are programmed with the schema go-to-goal, which commands the robots to move from the current position to the goal position. To use this schema, robots need to know their relative position with regard to the goal. However, not all the robots have the sensing capabilities to determine their relative position and, thus, the capability of navigating independently. Therefore, some robots need help from more intelligent robots to get the information they need in order to fulfill the assignment. As can be seen in figures 18, two robots (first group in green) move to reach the goal on the left, while other three robots (second group in blue) move to reach the position on the right. The other two robots left (third group in red) wait at the starting position for help. Once one of the teams has completed its task, the leader robot (more capable robot) of that team goes back to collaborate with the other two robots that still need help. This task allocation problem continues with new random position being assigned and robots collaborating to achieve their tasks.



(a) (b) (c) (d)

Figures 18a-18b-18c-18d: Models of task allocation problem using ASyMTRe, retrieved from L.E. Parker and F. Tang (2006)

In F. Tang and L.E. Parker (2007), the ASyMTRe-D algorithm is combined with a market-based task allocation algorithm. This combination is used to allow the robots to perform both strongly cooperative (or ST-MR-IA) tasks and weakly cooperative (or ST-SR-IA) tasks within the same application. More specifically, the authors apply the ASyMTRe-D algorithm to solve strongly cooperative tasks and the market-based algorithm to fulfill the weakly cooperative tasks. They demonstrate the feasibility of this algorithmic integration using simulation. The under-research simulation scenario includes four robots and five boxes, of which three are large boxes that require the application of a strongly cooperative solution (ASyMTRe-D) and two small boxes that require the application of a weakly cooperative solution (market-based). The simulation results prove the success of coupling the two mentioned algorithms to provide robots with the ability to generate both strongly and weakly cooperative solutions within one multi-robot application.

Summary. Concluding, micro modeling focuses on the analysis of specific tasks, behaviors, or communication and control architectures of robots. Thus, micro-models are implemented to test the accuracy of algorithms or communication and control architectures. The model of J.P. Desai et al. (2001) shows how a team of robots can be controlled and coordinated within a formation using a leader-follower algorithm and graph theory. Z. Wang and M. Schwager (2016) and B. Hichri et al. (2016) demonstrate in their models the coordination of the actions of multiple robots to transport parcels of different weights and shapes. The NetLogo model of D.L. Martínez and A. Halme (2016) depicts the coordinative ability of multiple robots to scan their surroundings and mark their explored space. Regarding path planning and obstacle avoidance, F. Duchon et al. (2014) shows the validity of Basic Theta* and Phi* algorithms to allow robots to search in every angle, and eliminating the dependency of a grid. The model of C. Cai et al. (2007) demonstrates the suitability of combining path planning and collision avoidance algorithms. In their simulation, the authors show how multiple robots can predict their safe trajectories by assigning priorities to other robots and static objects. Finally, L.E. Parker and F. Tang (2006) and F. Tang and L.E. Parker (2007) validate the ability of robots to perform simultaneously strongly cooperative and weakly cooperative tasks, using the ASyMTRe algorithms for complex tasks and market-based algorithms for simple tasks.

3.2.2.2 Macro-modeling

As observed in Paragraph 3.3.2.1, micro-modeling is leveraged when the aim is to test the fitness of algorithms or design methods. In comparison to micro-modeling, macro-models are not developed to assess an algorithm or a design method, but to analyze the collective behavior and performance of the entire system. Therefore, when building macro-models, the objective of a designer is typically to demonstrate the potentialities and/or limitations for the application of an MRS in a certain domain. Intuitively, a macro-model can include more than one algorithms and design methods, as it does not concern with the study of single collective behaviors, but with the analysis of system behaviors. For this reason, we now elucidate three examples of macro-models found in the literature. The reported instances of MRS have all applicability in a logistics scenario.

Example 1: MRS for transportation tasks

In a recent research, A. Farinelli et al. (2017) investigate the deployment of an MRS in warehouse logistics operations. Specifically, the authors examine coordination techniques for a set of robots involved in the transportation of materials from loading to unloading gates in a simulated warehouse environment. The key problem, they have attempted to solve is to make robots select individual decisions that lead to optimizing a system-wide objective function. Therefore, the authors start out solving a DCOP (Distributed Constrained Optimization Problem) task assignment problem using a max-sum algorithm, and then develop a macro-model to evaluate the capability of robots to maximize the key performance indicator, i.e. *task throughput* (number of objects transported per unit of time). The main issue in this research is for robots to avoid accessing the same loading gates and to avoid interfering with other robots in the shared spaces connecting the loading gates to unloading gates. Each robot estimates the impact that its actions has on the global optimization function, and consequently chooses the assignment that maximizes it. Comparisons between the max-sum algorithm and other heuristics algorithms are made under different scenarios. First, the authors use simulation to validate the application of the different task assignment algorithms to solve the DCOP problem in a specific logistics scenario. Subsequently, the aim of the simulation shifts to evaluate the system-wide performance of the MRS. Seven scenarios are implemented by varying the input parameters, being the number of loading/unloading gates, number of robots, number of transportation tasks. Every simulated experiment finishes when all orders are fulfilled. The warehouse topology used in the experiments represents the typical situations handled by logistics operators, where the number of loading gates is higher than the number of unloading gates. The key performance indicators evaluated in each experiments are (1) the average number of robot interferences, (2) the maximum number of robot interferences and (3) the average task completion time (ATCT), which is the average time required by robots to accomplish a given transportation task.

The simulation results show that the max-sum algorithm does not provide significantly better results with regard to the average robot interactions, in comparison to other heuristics, but it is able to reduce intensively the maximum number of interferences and the ATCT. Strictly speaking, in the simulated environments, the number of robots sharing the same route does not change to a significant extent; however, the number of critical situations, i.e. the number of times where too many robots share the same route, decreases drastically in comparison to other heuristics. Furthermore, the max-sum algorithms always incur lower ATCT, with robots completing a task in around 40 s in a scenario with five robots having a max speed of 0.5 m/s, 20 loading gates and 5 unloading gates with an average distance of 3 m between them. In addition, the max-sum algorithm scales better with the number of robots, e.g. with 10 robots and same number of loading/unloading gates, the ATCT reduces to 14.5 s, 10,37 % improvement in comparison to other heuristics. Thereupon, it can be concluded that the max-sum algorithm is capable of improving the task throughput of the MRS significantly.

In comparison to the micro-models described in Paragraph 3.3.2.1, this research does not only focus on the assessment of an algorithm, but also on the performance of system-wide operations (e.g. *maximization of task throughput*).

Example 2: Alphabet Soup

Alphabet Soup is an abstract model of the real-world problem of order fulfillment and assembly in a warehouse environment (C.J. Hazard et al., 2006). The alphabet soup simulation focuses primarily on the resource allocation problem for an MRS, but also on the coordination of multiple vehicles. This macro-model was designed to support the development of the Kiva System by providing answers to the research questions: Where to store shelving units in the warehouse? Which shelving units to bring to which picking stations? Which picking stations to provide SKUs? Which inventory stations to assign incoming SKUs? (R. D'Andrea, 2012). In this situation, however, buckets represent the shelving units; letters represent SKUs; the letter stations represent the inventory stations; the word stations represent the picking stations. In this way, the model describes the pick-pack-and-ship operations of distribution warehouses (Figure 19).

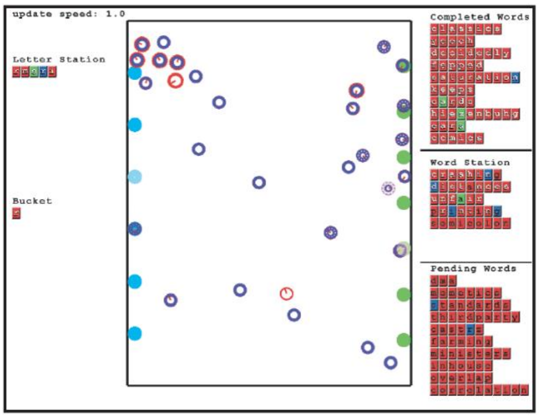


Figure 19: Alphabet Soup, retrieved from C.J. Hazard et al. (2006)

The moving units are called bucketbots, i.e. small robots that are used to move buckets from letter stations to word stations to accomplish the transportation tasks. In the letter stations, letters are allocated into buckets; in the word stations, letters are taken out from the buckets. Incoming letters are collected at the letter stations, which have a limited capacity of space to receive letters. Bucketbots bring buckets to the letter stations, to get the letters into the buckets. Buckets are then distributed to the word stations to construct a set of words. A word station takes a letter out of a bucket as long as this is required to complete one of its active words. When the word station completes a word, it puts it into the "completed words" list. Like for letter stations, word stations also have a limited capacity with regard to how many words they can contain. Bucketbots perform repeatedly the same actions: grabbing a bucket; releasing a bucket; accelerating; decelerating; and communicating with a letter station to take a letter from it or to place a letter into a bucket. Furthermore, robots can only pick up a bucket at a time, and they should not collide with other robots, otherwise they are stopped and penalized. Temporal actions are also included in the simulation, including the time that a bucketbot needs to pick up a bucket, the time that is required to remove a letter from a bucket, and the time that a word station takes to place a finished word to the completed words list.

The performance of the system is evaluated based on the following performance indicators: the number of completed words; total and average distance driven covered by bucketbots; number of collisions; average idle time and average capacity utilization of bucketbots; and idle time of letter/word stations. The results show that in order to have steady performance, the throughput of word stations must be balanced according to the throughput of letter stations, and the number of bucketbots required has to be big enough to keep low the idle time of letter/word stations and small enough to keep low the idle time of bucketbots.
The simulation experimental design consists of 25 word stations, 25 letter stations, 250 bucketbots, 850 buckets. Furthermore, robots have a diameter of 2 meters each and maximum speed of 4 m/s; buckets have a capacity of 40 letters; buckets can be picked up and released at 0.5 s; letters can be moved into buckets and from buckets to word stations in 5 s. Within this context, results feature good performance with regard to the time required by robots to construct words (around 20 s).

However, the minimal coordination implemented leads to high congestion, and consequently high number of collisions. Moreover, the task allocation is far from optimal, with too much time wasted by bucketbots to return buckets to letter stations for refill. According to the authors, the most challenging problems faced in the implementation of this system concerned with the task assignment problems, queuing and scheduling problems, i.e. when the bucketbots arrive at a word station, these need to be scheduled for deliveries. The authors suggest auction-based algorithms for the task assignment problems, while admitting that a wide variety of solutions exists to deal with the mentioned problems.

Example 3: Combining conventional path planning techniques with Swarm Intelligence Theories

D. Sun et al. (2014) propose a behavior-based multi-robot collision method to address the problem of efficiently coordinating a large number of robots performing transportation tasks in crowded logistic environments. This method combines traditional path planning algorithms with swarm intelligence techniques, found in nature. Each robot computes its optimal path using the A* algorithm and without considering the paths of other objects. During the execution time, each robot selects automatically the most appropriate behaviors to avoid collisions by detouring temporarily from the optimal route. Robots are configured with eight internal behaviors that they can select according to the situations at stake, and which are inspired by traditional traffic rules. The first basic behavior is "*FollowWayPoint"*, which implies that each robot follows its next waypoint in a grid until it reaches the destination. The second behavior is *"Avoid"* where a robot goes round an obstacle, i.e. another robot or a static object. *"Exhange"* is used to avoid frontal collisions, where two robots change trajectories to avoid hitting each other. *"GoThrough"* is activated when a side collision is forthcoming. In this situation, two robots are going through an intersection from different directions; therefore one robot has to wait for the other to pass before moving on. A *"Dock"* behavior entails that a robot, once reached a station, starts docking at the station. The robot that has to wait when the other robot drives round or this is on a head-on collision proximity prompts a "*WaitKeepDistance*" behavior. The robot that waits in an intersection activates a *"WaitForGoThrough"* behavior. Ultimately, when another robot is already docking to a station, the successive robot activates a "*WaitForDocking"* behavior to wait for the first robot to complete its operations.

Simulation is used to evaluate the effectiveness of the proposed method by comparing it with other methods. The objective of the simulation is to identify the method that provides the minimum completion times for the given delivery tasks, being this the only performance indicator observed. Two input parameters are varied in the experiments: number of robots and number of deliveries. The analysis of the simulation results shows that the developed approach provides higher benefits in terms of quality solutions (minimum completion times), scalability and requires less time to compute paths in comparisons to other conventional path planning algorithms. Furthermore, the results show that the average velocity of robots (m/s) is higher using this method, because robots require less time to compute their paths and solve conflicting trajectories; thus, robots are able to reach rapidly their goal stations.

Same as example 1, the authors of this research have the primary objective to assess the fitness of the proposed algorithm for the motion-planning problem in large robot teams. Nevertheless, in comparison to micro modeling, the research does not merely target to analyze specific collective behaviors of robots, but the authors put their focal point on the system behaviors, trying to optimize the global performance by means of the developed strategy. Therefore, this model belongs to the category of macro modeling.

Summary. Macro modeling concerns with the analysis of system (macro) behaviors, rather than with the study of specific robot behaviors. In this sub-paragraph, three instances of macro-models have been reported, which investigate the application of MRS in logistics scenarios. From the first analysis, we can state that the max-sum algorithms for the task allocation problem is revealed appropriate to reduce the maximum number of robot interferences and the average completion task time. Furthermore, the simulation results evidence the effectiveness of using multiple robots to execute transportation tasks of materials from unloading to loading gates in a warehouse. From the second analysis, the Alphabet Soup model provides us with a high-level representation of transportation tasks of letters, which are used to construct words at the word stations. Furthermore, the key performance indicators employed in the simulation fit with the analysis of performance of a sorting system. The simulation results also demonstrate that the number of robots required for a transportation task needs to be well calibrated in order to increase the utilization rate of the picking and dropping stations, and the utilization rate of robots. Finally, this analysis demonstrate how difficulties in the simulation of a similar model may be caused by task assignment, queuing and scheduling problems. From the third analysis, the developed model proves the suitability of using a reactive control scheme to deal with collision avoidance problems. This control architecture bestows scalability, robustness and, in this case, quality solutions on robotic systems. During the delivery tasks, robots have eight behaviors they can prompt to resolve potential conflicting trajectories and find collision-free paths. The analysis of simulation results show that, using this strategy, robots take less time to compute their collision-free paths. Altogether, the analyzed researches will contribute to the realization of a macro model featuring a large number of robots executing transportation and sorting tasks of parcel in a sorting hub.

3.3 Thesis contributions

The contributions of this project find placement in the macro-modeling and algorithms fields with complementary additions in the design methods (Figure 20). In this research, a holistic approach is taken to evaluate the effectiveness of an application of a MRS in a sorting hub. Specifically, the main contribution of this thesis will be a high-level macro-model featuring the use of multiple autonomous robots for transportation and sorting tasks. This application covers all the examined domains of MRS, being motion coordination, mapping and localization, path planning and collision avoidance, task assignment. Moreover, this application includes another relevant topic that requires investigation, which is the queueing domain. However, this research focuses mainly on finding a solution for the task assignment and motion coordination of robots. In particular, tasks in this application can be separated in ST-SR tasks, i.e. tasks that do not require the cooperation of robots, and ST-MR tasks, i.e. tasks that require the cooperation of robots. In order to solve the task assignment problem, we will develop a method for the dynamic switch of robots' behaviors when facing different types of tasks. Furthermore, ST-MR tasks require the collective motion of robots to transport materials from origin to destination. In fact, robots need to create a formation and retain it until the end of the task. A solution will be provided for this problem. Nevertheless, the other domains of MRS are not overlooked and decentralized algorithms are implemented for their resolution.

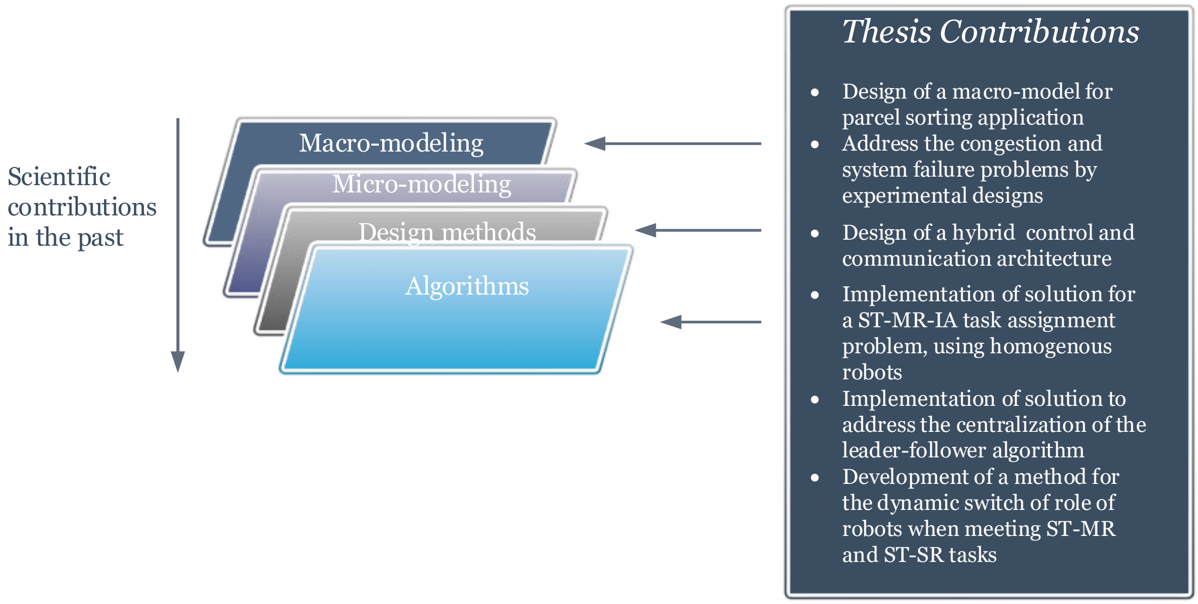


Figure 20: Thesis contributions framework

With regard to the control and communication, robots are considered autonomous entities capable of reasoning and facing their situation based on pre-defined behaviors. Therefore, robots are all designed with a decentralized control, which give them the decision authority. Communication is for the most part decentralized, with robots communicating with each other within short communication ranges. Nevertheless, some centralized schemes are necessary and advisable to improve the performance of the system. A hybrid control and communication architecture is, thus, used to increase the system robustness, scalability, flexibility and performance.

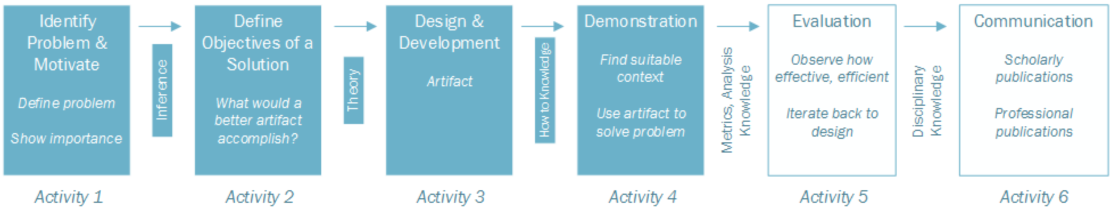
Experimental designs are implemented to evaluate the impact of collaborative transport of robots on robustness, scalability, flexibility and performance of the system. A thorough analysis of congestion level and disruption will show this impact. Multiple scenarios are developed to fulfill these knowledge gaps. Furthermore, three system configurations (or design choices) are explored to find the best configuration, i.e. the configuration that provides better system performance, scalability, flexibility and robustness.

Conclusions

Overall, we can conclude that different classes of MRS exist, namely Collective, Cooperative, Collaborative and Coordinative. These classes can be discriminated based on the awareness or unawareness of robots of the existence of other robots, the nature of the tasks that can be individually or jointly performed, and the fact that the actions of robots can or cannot support the actions of other robots. The system studied in this research belongs to two categories, being Coordinative and Cooperative robotics. In fact, in this system, part of the robots do not have common goals and their work do not benefit from the work of others. These robots perform single transportation tasks. Other robots, however, share common goals and need the support of other agents to execute a task, thus exhibiting cooperative behaviors.

The most challenging domains for an indoor application of MRS are communication and control schemes, motion coordination, localization and mapping, path planning and collision avoidance, task allocation. A taxonomy of MRS has been presented, which divides the field into two branches, namely analytical/low level and theoretical/high level. In the analytical/low level, we have investigated few algorithms for every domain of MRS and illustrate some control and communication design methods that can be used in MRS. Subsequently, the theoretical/high level field has been observed, with a clear distinction made between micro- and macro-modeling. Micro modeling is concerned with the study of specific behaviors, whereas macro modeling focuses on high-level systemic behaviors. In this research, we mainly focus on the macro modeling area, with contributions also made in the algorithmic and design method fields. In the next Chapter, we describe the conceptual model, the use of agent-based simulation for the implementation of this model, the implemented model and the performance indicators leveraged to analyze the results of the simulation.

4 | Model development



Introduction

In the previous Chapters, we have investigated the literature to create a theoretical foundation of important notions upon which we produce our practical results. In this Chapter, guided by a Modeling Development flowchart, we detail the steps required for creating a model of a new parcel sorting system.

Just like actors starring in a drama, we first decide the main characters (*system elements*); next we assign them with a script (*define interactions between elements and process events*); and subsequently we describe how they should move on stage (*decide and formalize mathematical or verbal algorithms).* These activities constitute the model conceptualization part, which can be seen as describing the narrative of the story. In order to evaluate how the system performs, we define the inputs and outputs (or performance indicators) for this application. Afterwards, this narrative needs somewhat to be put in action. In modeling, this phase, called model implementation, is done through computer implementation and programming. For the implementation of this model, we have decided to employ NetLogo, an open-source java-built tool. Few verification tests are reported to build confidence into the results of the simulation. Two experts are interviewed in order to validate the outcome of the model. The results of the interviews are that the model represents sufficiently parcel sorting operations and experiments need to be designed appropriately in order to respond the investigated research questions. This Chapter covers activities 3 and 4 of the DSRM, being Design and Development and Demonstration.

This Chapter is structured as follows. In paragraph 4.1, we briefly describe agent-based computing and make comparisons between agent-based modeling and multi-agent systems. This paragraph is used to collocate our model into the broad spectrum of agent-based computing. In paragraph 4.2, we describe the modeling development steps, using the traditional explore-model-test-evaluate framework for modeling and simulation. In paragraph 4.3, the model conceptualization phase is broken down into four parts. In sub-paragraph 4.3.1, we refine the requirements and constraints for the design of a multi-robot parcel sorting system. In sub-paragraph 4.3.2, the system elements are identified. In sub-paragraph 4.3.3, the interaction among system elements are described. In sub-paragraph 4.3.4, the process events performed by robots are detailed out. In sub-paragraph 4.3.5, the algorithms used to put in motion the actions of system elements are defined. In paragraph 4.4, the performance indicators needed to quantify the KPIs for this system are pointed out. In paragraph 4.5, we briefly describe how we have implemented the model in NetLogo. In paragraphs 4.6 and 4.7, the model verification and validation are performed. Finally, in the conclusions, a table with the assumption made for building the model are listed.

4.1 MAS and ABS for AI

Agent-based computing, a vibrant and diverse scientific domain of information technology, emerged in the 1990s. Since then, it has widely expanded and progressively evolved into a broad spectrum of communities (M. Niazi and A. Hussain, 2011), including Agent-Based Modelling (ABM) and Multi-Agent Systems (MAS). The shared viewpoint about the notion of "agent" has generated confusion regarding the differences among the diverse communities. In fact, these disciplines of agent computing do not present distinct properties with regard to the definition of agents. An agent is an encapsulated[1] software entity operating independently, having some control over its actions and internal state (*autonomy*); communicating with other agents in order to execute its tasks (*social ability*); reacting flexibly to inputs from the environment (*reactivity*); and exhibiting goal-directed behaviors (*pro-activeness*). Therefore, there is consensus upon the fact that autonomy, ubiquitous computing, ubiquitous communication, adaptation, responsiveness are key features of all agent-based disciplines (M. Wooldridge, 1998).

However, the notion of agent is the only overlapping aspect of the diverse disciplines of agent-based computing. In fact, ABM and MAS present clear differences that are

[1] Precisely identifiable, with well-defined boundaries and interfaces

important to highlight in order to give a precise collocation to our work into the spectrum of agent-based computing.

ABM are employed with the aim to detect emergent behaviors, using a bottom-up approach. Thus, these models are used to respond the question *What happens when…?* rather than modeling a system with a desired goal in mind. ABM is therefore most appropriate for domains of natural and social sciences, to gain a better understanding of global emergent phenomena in complex social systems.

Instead, in MAS the scope is typically to solve a given problem or achieve a certain state, typically using a top-down approach. MAS is used in Artificial Intelligence to study groups of intelligent, problem-solving agents. The use of MAS for the study of AI problems is referred as Distributed Artificial Intelligence (DAI). The emphasis when implementing MAS models is on answering questions like *How can I make a …?* or *Can the system do this?* instead of replicating behaviors of agents in real-life situations to observe their emergent, often hidden, properties (K.H. van Dam, I. Nikolic and Z. Lukszo, 2013). Same as ABM, MAS modelling also finds application in dynamic multi-agent domains, such as in environments where numerous agents must interact and operate within rapidly changing conditions. However, unlike ABM, MAS models enable software engineers to find solutions to physical problems that are otherwise unachievable, due to the high complexity involved (M. Luck et al., 2005). For example, MAS models are adopted to address issues such as balancing reactivity and deliberation in control and communication design methods, developing cooperative strategies, e.g. resource allocation problems or team formation patterns, and managing large-scale technical systems.

Evidently, the model we develop lies in the MAS domain, as its scope is to design a multi-robot system executing parcel-sorting operations, develop cooperative strategies, balance centralization and decentralization schemes and improve flexibility, operational robustness and efficiency of parcel sorting operations. However, in this project, we exploit the benefits of Agent-Based Simulation (ABS) to gain better insight into the application of a MRS for parcel sorting. It must be noted that ABM and ABS are not identical notions. ABM create computer models composed of agents and objects operating in an environment through simple local rules. ABM are implemented in a computer software as agent-based simulations. ABS is then used to test the impact of changing variables on the global system behavior. The model we develop is subsequently implemented in NetLogo, a java-built ABS software. P. Davidsson (2000) argues that ABS is suitable for simulating scenarios in technical domains since it supports structure-preserving modeling of simulated reality, simulation of parallel computations and dynamic simulation scenarios. Furthermore, the author explains the advantages of ABS in comparison to traditional Discrete Event Simulation (DES). In short, in ABS it is possible to increase or reduce the number of agents during the simulation without suspensions, thus allowing for scalability and dynamic simulation scenarios. Furthermore, ABS facilitates distributed computation, thus enabling a simple implementation of large-scale agent-based models. However, ABS uses more computation and communication; therefore, it may lead to slower simulations.

4.2 Modeling Development Flowchart

In the previous paragraph, we have collocated the model in the MAS domain and anticipated the use of ABS to simulate the developed model. Hereinafter, we describe the modeling process adopted to construct an abstract representation of a parcel sorting system using a multi-robot approach.

Figure 21 gives an overview of the Modeling Development Flowchart adopted in this project, which involves four steps, namely explore, model, test and evaluate. The Modeling Development Process starts with the exploration of the context, in which the sorting procedures take place. This phase was performed in Chapter 2, from which we have comprehended the problem experienced in the sorting terminals and we have defined the requirements and constraints for the new design. Furthermore, the exploration is enriched with the analysis of a new warehouse automation trend that involves the use of multi-robot system for the application of logistics operations. Therefore, we have provided a thorough investigation into the field of multi-robotics to develop a theoretical knowledge required to design the new system. After the exploration phase, a conceptual model is developed, which entails a logical (algorithmic) or verbal representation of the new sorting design. This involves the analysis and decomposition of the sorting practices into a collection of agents, objects, internal states, rules (relevant behaviors), and external world in which interactions occur. Subsequently, mathematical and logical algorithms formalize how the agents are supposed to operate in the environment, i.e. how agents respond to certain inputs to produce desired outputs. The next phase is the development of the computerized model through computer programming and implementation. Hence, the computerized model is the implementation of the conceptual model on a computer. To ensure that the computer implementation of the conceptual model is correct, model verification is executed. Model verification allows answering the question "*Is the model right?*", thus checking if the implemented model works in accordance to the specifications given in the conceptual model, and indirectly verify if the specifications of the conceptual model are also correct. When the computerized model is verified, still another question arises, being "*Is the right model?*". This question is answered through the validation phase, which determines if the model's outcome is sufficiently adequate to provide the necessary responses on problems regarding real-world situations (R.G. Sargent, 2005).

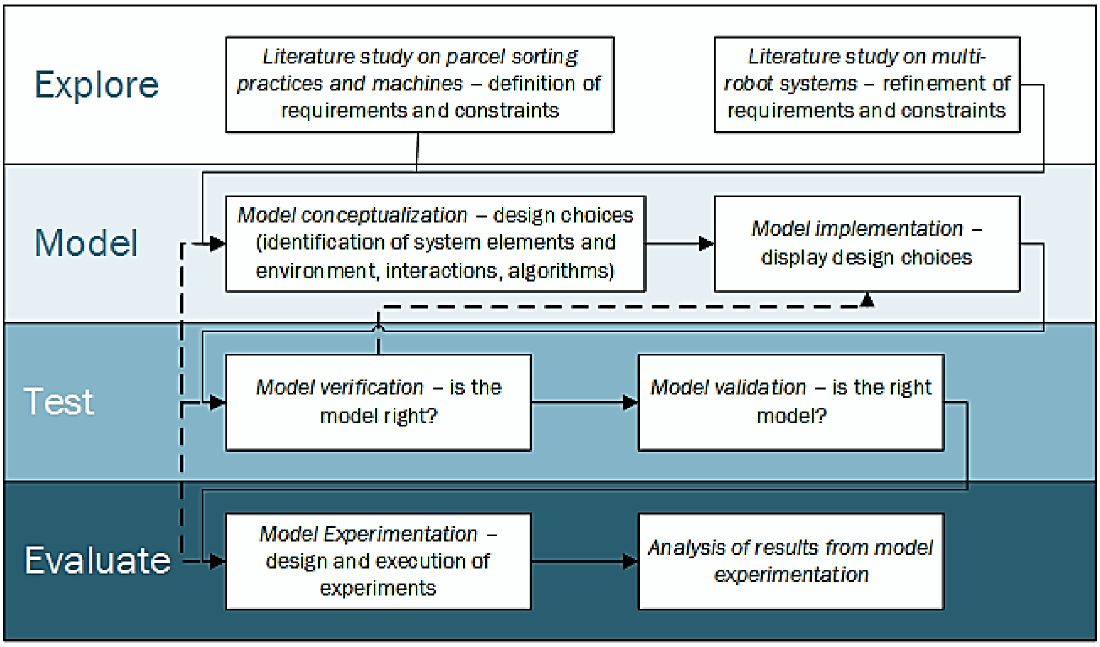


Figure 21: Overview of the Modeling Development Flowchart

Earlier, we have formulated the problem to address by means of modeling and simulation. Therefore, the developed model will serve as a problem-solving and decision-making instrument. Therefore, modeling and simulation are a means to design a new sorting solution, rather than as a means to analysis a current situation. Model experimentation is used to evaluate the results of the new system on the KPIs, being system effectiveness, congestion and fault tolerance. Furthermore, this will be used to make decisions concerning the most appropriate traffic configuration, the communication and control schemes, and the algorithms to use for motion coordination, localization and mapping, path planning, collision avoidance and task allocation. In particular, a special focus will be made on addressing the problem of transporting heavy, high volume parcels, other than small parcels, using cooperative behaviors. Ultimately, it will attempt to address the impact of cooperative behaviors on fault tolerance and performance.

In the next paragraph, we develop the conceptual model showing the series of operations that robots are supposed to perform in the environment and the flow of interactions agents-to-agents and agents-to-objects ensuing from environmental inputs.

4.3 Model Conceptualization

As stated earlier, the conceptual model is the phase where the problem entity (i.e. MRS for parcel sorting) is decomposed into its main actors, the storyline is described and a personal script is provided to each actor. Adapted to the systems engineering world, this entails:

i. Identifying the system elements, defining their internal states and rules, and defining the environment in which the system elements are placed;
ii. Defining the interactions between system elements and between system elements and environment;
iii. Defining the process events, i.e. what system elements are supposed to do;
iv. Deciding upon and formalize algorithms to use to address the problems explained in Chapter 3.

4.3.1 Fine-tuning requirements and constraints

In Chapter 2, paragraph 2.5, we have described the functional / non-functional requirements and constraints for a generic sorting system. The new multi-robot sorting system must execute the basic functionalities required by the sorting tasks, such as transport and sort parcels. In comparison to traditional sorting systems where parcels of different size and weight travel on conveyor belts, robots need to carry out different tasks using cooperative and non-cooperative behaviors, as it will be extensively explained in this paragraph. In order to reduce the complexity of the model, we have decided not to focus on certain functional and non-functional requirements of sorting systems, such as on maintenance, energy consumption and economic feasibility of a multi-robot system. In this research, we have put our focal point on the development of solutions for the problems illustrated in Chapter 3, such as motion coordination, resource and task allocation, path planning and collision avoidance and localization and mapping. Furthermore, we are only partly considering the operations performed by human operators (or their robot surrogates). In this research, we are not considering the time an operator takes to move full containers to the exit points and replace them with empty containers. While the temporal intervals for loading and unloading operations are included. Concerning with the non-functional constraints, the objective of this research is to evaluate the system effectiveness, robustness and congestion of the system. Accordingly, we want to assess if the new system shows off the benefits of flexibility, scalability, fault tolerance, throughput and utilization rate. Moreover, although we will not test the system in multiple different sorting environments, we can infer some conclusions regarding the reusability/adaptability of the new sorting system. Furthermore, the economic feasibility of the new system is neglected. Therefore, we are not investigating the investment cost required by the new system, which hinges on three main determinants, being investments on hardware, software and infrastructural extensions (e.g. on static cameras or beacons). Consequently, we are also neglecting the operational costs, such as energy and maintenance cost.

Table 9. Constraints for multi-robot parcel sorting system

Parameter	Unit	Defined values
Sorting Center capacity	[parcels / hour]	5000 - 8000
Number of sorting directions (destinations)	[]	50
Floor area	[m^2]	3750
Unloading operator capacity	[parcels / hour]	800
Loading operator capacity	[parcels / hour]	900
Parcel size distribution	[cm x cm]	50 x 50 and 180 x 80
Parcel weight distribution	[kg]	2.5 and 30
Parcel destination distribution	[]	uniform
Robot max speed	[m/s]	2
Robot max acceleration (unloaded)	[m/s^2]	1
Robot deceleration	[m/s^2]	5
Robot length	[m]	0.9
Robot width	[m]	0.6

In Chapter 2, we have also defined the constraints for a sorting system, showing the typical thresholds used when designing conventional sorters and min and max intervals. In this research, we have defined specific constraints to carry out the initial test of the new system. Table 9 shows the thresholds we have agreed upon with Prime Vision. In comparison to the constraints defined in Table 5 (Chapter 2), we have ruled out the walking speed of human operators, the container exchange distance and the max number of parcels onto containers. As already stated, we are not taking into consideration the transportation and replacement of full containers with empty containers. Therefore, we do not need to include the mentioned constraints. In addition, we need to include extra constraints that relate to the inner characteristics of robots. Robots must be designed with limited dimensions (90x60 cm) in order for the system to be economical, to increase the number of robots (increasing the throughput) and increase their agility and to reduce the energy consumption. Further, robots can have a maximum speed of 2 m/s. The maximum acceleration is of 1 m/s^2, when robots are not transporting parcels. The maximum deceleration is of 5 m/s^2, which is the deceleration required to stop the motion of robots moving at maximum speed. Furthermore, the sorting system will be placed in an area of 3750 m^2, which corresponds to a sorting terminal of small-medium dimensions. In this floor area, there are 50 sorting directions, while the number of pick-up buffers are around half this number. Under these specifications, the goal is to achieve a throughput of between 5000 and 8000 parcels per hour.

4.3.2 System Elements and Environment

In Chapter 2, we have identified the root elements of every traditional parcel-sorting hub. These elements are *parcels*, *pick-up* (or load) *buffers, drop-off* (or unload) *buffers, inbound unit load device* (ULD), *outbound unit load device, inbound trucks, outbound trucks* and *conveyor belts* (sorting machines).

Not all the elements of our system line up with the root elements of a traditional parcel-sorting hub. First, robots replace conveyor belts, thus becoming the new sorting devices. Second, we stated in Chapter 2 that this research focuses on the operational level of sorting systems, excluding truck scheduling, destination assignment and layout planning, which are at strategic level. Consequently, we can exclude inbound and outbound unit load devices, and inbound and outbound trucks. Therefore, the system elements of the multi-robot parcel-sorting system are *robots*, *parcels*, *pick-up buffers* and *drop-off buffers*. Thus, this system corresponds to a tuple (R, P, U, O), where:

- $R = \{r_1, \dots, r_i\}$ set of Robots
- $P = \{p_1, \dots, p_j\}$ set of Parcels
- $U = \{u_1, \dots, u_l\}$ set of Pick-Up Buffers
- $O = \{o_1, \dots, o_m\}$ set of Drop-Off Buffers

Robots are the only agents in this system, since they are dynamic software entities, capable of flexible and (semi) autonomous actions, meaning that they have some control over their internal state and behaviors and they can respond flexibly to actions of other agents or objects. Furthermore, we consider the use of homogenous robots, with identical capabilities and traits, and relatively simple, with limited hardware capabilities.

The basic internal states and rules of robots are displayed in Figure 22. The individual agents can have an unloaded or loaded status, corresponding to whether they are transporting a parcel or not. Robots perform simple actions, collecting parcels from pick-up buffers and unloading parcels at drop-off buffers. Therefore, every robot has a *"my pick-up"* and *"my drop-off"* state that contain information with regard to the buffers where parcels can be requested and delivered. Another basic state corresponds to the *"my parcel"* condition that ensures the assignment robot-parcel is a 1-to-1 assignment, meaning that each robot can receive one and one only parcel from a pick-up buffer. Before requesting a parcel inside a pick-up buffer, robots position themselves in a queue following a FIFO (first-in-first-out) scheme. Therefore, robots that arrive first in a queue are the first that leave the queue, entering the pick-up buffers.

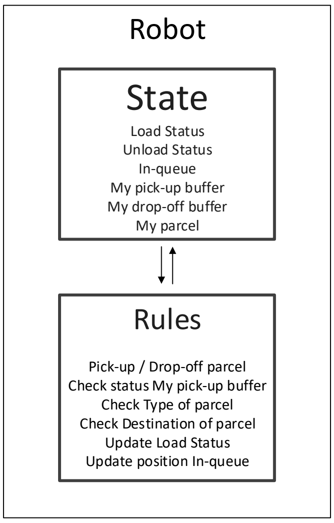
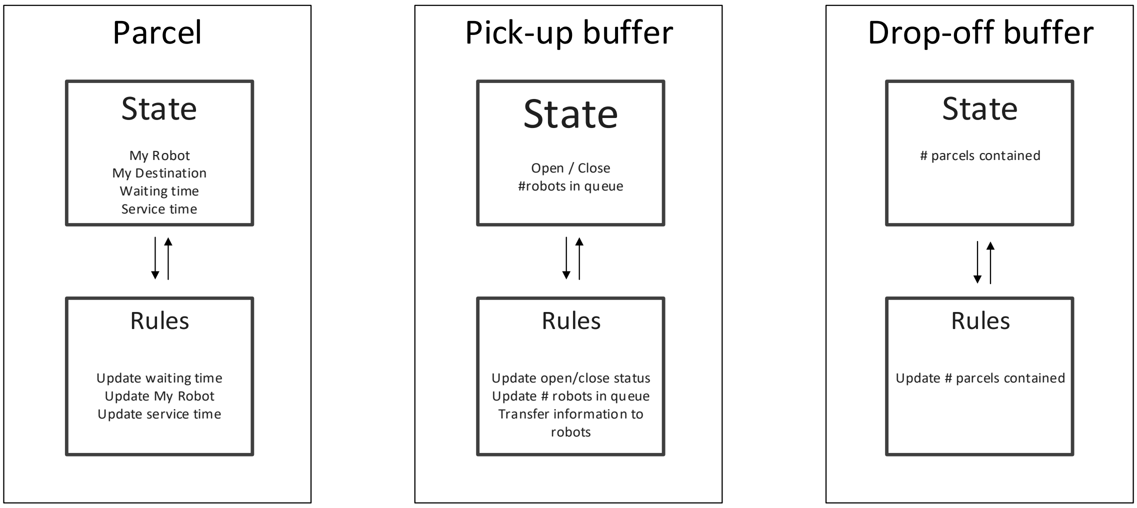


Figure 22: Robot's internal states and rules

Rules are the basic internal behaviors performed by every robot that lead to state changes together with interactions with other agents or objects. As stated earlier, robots only execute simple tasks. They place themselves in queues, updating their positions every time a robot in front moves ahead. When they arrive in front of a pick-up buffer, robots check if another robot(s) does not occupy the buffer. If buffers are free, robots access them, requesting a parcel. At this point, robots wait a number of seconds to load the parcel and transport the parcel to the appropriate destination. These are the basic actions performed by each robot in the system. In addition, robots can solve more complex tasks: avoiding collisions with other robots, planning their best paths, improving the distribution over pick-up buffers and transporting heavy and high volume loads. Later in this Chapter, the algorithms used to address these problems are pointed out.

Robots are the only dynamic agents of the system. The other system elements correspond to static objects like parcels, and structural elements, like pick-up and drop-off buffers. Dissimilarly to robots, *parcels*, *pick-up buffers* and *drop-off buffers* are software objects, i.e. fixed and static entities that perform actions only when invoked (Figures 23a – 23b – 23c).



(a) (b) (c)

Figure 23a-23b-23c. Internal states and rules of parcels (a), pick-up (b) and drop-off buffers (c)

Parcels randomly arrive at a pick-up buffer, i.e. we cannot predict which parcel is at which buffer. These static objects stand still at pick-up buffers, waiting for a robot to load and transport them to destination. The robot-parcel assignment is a 1-to-1 assignment, therefore, as for robots, parcels also have a *"my robot"* internal state. This ensures that each parcel is assigned to one and only one robot, thus eliminating disputes between robots that select the same parcels. Furthermore, every parcel has a *"my destination"*, which gives information to robots regarding the drop-off buffers to follow in order to deliver the parcels to the appropriate containers (sorting operations). *"Waiting time"* and *"Service time"* refer respectively to the time a parcel waits at the station before being collected and the time a robot uses to transport the parcel to destination. The internal rules of parcels correspond to the actions executed by parcels to change their states. Therefore, the only actions performed by parcels are updating their waiting / service time and updating the my robot state.

Robots move towards the pick-up buffers to perform their transport tasks. However, these buffers are not always open, i.e. when another robot is inside the pick-up buffer, other robots have to wait before accessing said buffer. The *"Open / Closed"* internal state of pick-up buffers ensures that a limited number of robots enter these areas, while the others wait patiently in queues. Each pick-up buffer counts the number of robots in queue, to observe the distribution of robots and improve the assignment of robots to pick-up buffers. Apart from updating open/closed state and the number of robots in queue, another essential rule executed by pick-up stations is the *"Transfer of knowledge / information"* to robots. This communication enables taking away overcrowded circumstances inside and outside the stations (e.g. balanced distribution of robots in queues). Drop-off buffers correspond to the destination of parcels and, in real-life, to containers where parcels are accumulated. Thus, the only internal state of drop-off buffers is the *"number of parcels contained"* and the only rule is the *"update of number of parcels"*.

In A. Farinelli et al. (2017), we have seen that in traditional distribution / sorting centers the number of pick-up buffers is lower than the number of drop-off buffers. In traditional sorting centers, in fact, the number of drop-off buffers is almost three times the number of pick-up buffers. For this reason, given the large number of drop-off buffers, additional queues outside drop-off buffers can be removed. This is done in consideration of the fact that robots very rarely arrive at the same drop-off buffer at the same time, as we show later in the model implementation phase. In case they arrive precisely at the same time at the same drop-off buffer, the late robot waits outside the drop-off buffer for the other to unload the parcel.

The environment is defined as all system components that cannot be influenced by other system elements (e.g. agents / objects), but that can affect the actions of other system elements. Therefore, the environment is that part of the model composed of exogenous variables, i.e. independent variables that influence the model without being affected by it in return. In our model, the environment incorporates pre-determined features, such as the position of the network of structural elements and of robots, and dynamic variables. These dynamic not-predefined changes include the random distribution of parcels across pick-up buffers and the random assignment of destinations to parcels. Comparable to real-life sorting situations, the distribution of parcels across pick-up buffers cannot be anticipated unless pre-sorting operations are prompted. Nevertheless, pre-sorting operations imply the use of another sorting system upfront the final sorting operations. Therefore, pre-sorting operations are uneconomical and often avoided by parcel logistics operators. The stochastic assignment of destinations to parcels influences the service time of robots, since longer or shorter distances to reach the appropriate containers may be travelled according to the destination of parcels.

In the next paragraph, the interactions agent-to-agent, agent-to-object and agent-to-environment are illustrated and embedded in an overarching framework.

4.3.3 System Elements and Environment Interactions

Interactions are the ways through which system elements communicate between each other to affect their behaviors/rules and, consequently, change the values of their internal states.

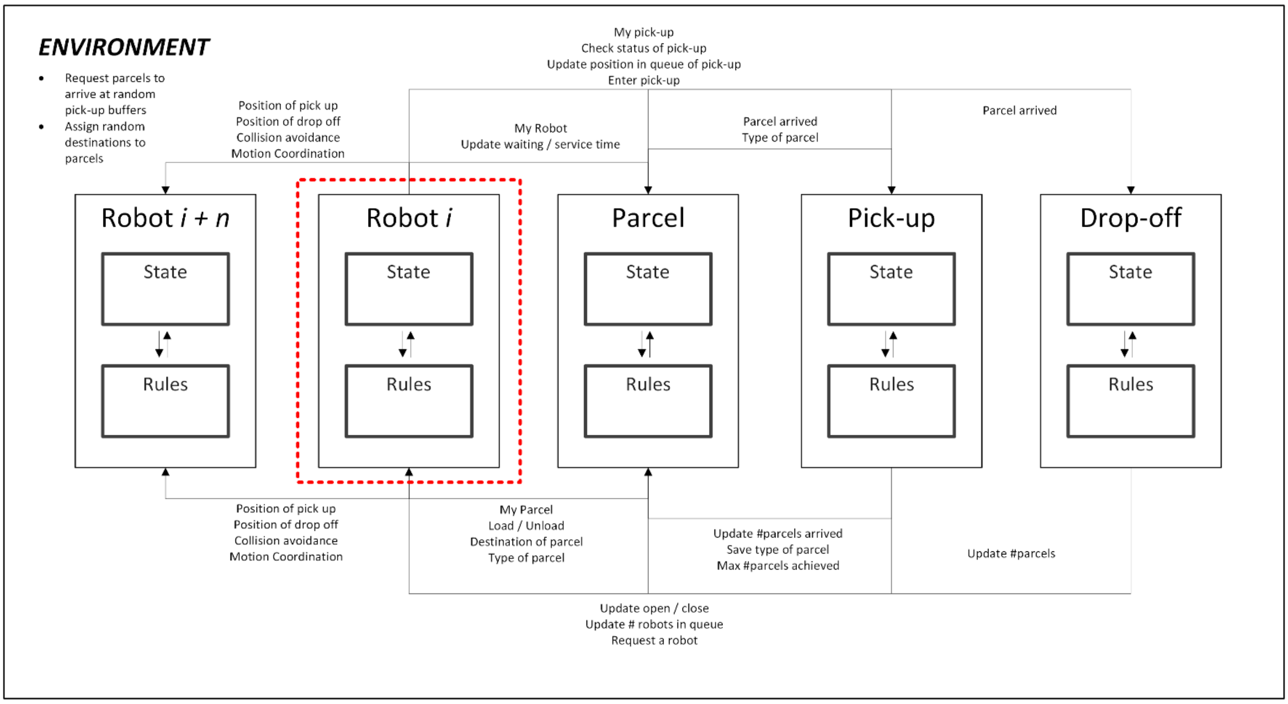


Figure 24: Communication agents-objects-environment

The environment can interact with agents or objects in order to alter their behaviors and states. Figure 24 shows the information sharing between system elements and environment.

In this system, robots communicate with other robots (robot-to-robot), with parcels (robot-to-parcel) and with pick-up buffers (robot-to-pickup). Robot-to-robot communication enables robots to step up the map-building process by sharing data with regard to the position of the pick-up and drop-off buffers identified in the environment (see 4.3.4 for process phases). Furthermore, in the event of potential conflicts in the paths of different robots, robots communicate in order to dodge other robots. Additionally, the communication robot-to-robot allows the cooperative transport of parcels whose load or size is excessive for an individual robot. In those cases, robots create a formation and coordinate their motion by communicating their speed and steering angle between the members of the formation.

Robot-to-parcel communication allows robots to reserve a parcel, altering the value of *my parcel* state. Robots require parcels to provide information regarding their type (e.g. high volume, heavy parcels). Based on the information obtained, robots decide whether they should act individually or collectively. When a parcel is loaded on a robot, this robot registers the presence of the parcel and modifies its status from *unloaded* to *loaded*. Loaded robots are ready to leave the pick-up buffers, but before doing so, they request the destination to their parcels. Next, parcels are transported to the appropriate destinations and the status of robots resets to unloaded. Parcels also communicate with robots (parcel-to-robot) to synchronize the *my robot – my parcel* assignment. When a parcel is claimed, it takes note of the robot in charge to deliver it. This is also a useful information for tracking parcels and avoiding their misplacement. Furthermore, when claimed, parcels update their waiting time and initialize their service time. Once arrived at destinations, parcels are unloaded from robots and the service time of parcels is stopped.

The communication robot-to-pickup enables robots to acquire information regarding the *open/closed* state of the buffers. Just like cars at red traffic lights, closed buffers enforce robots to wait in queues before getting the green light and moving inside. This information-sharing allows controlling and regulating the incoming flow of robots into the buffers. Furthermore, the communication robot-to-pickup can be used to adequately distribute robots across the buffers (*resource allocation*), eliminating the possibility to disregard or overburden certain pick-up buffers. In addition, robots update their positions in the queue, by communicating with the buffers. When the preceding robot has entered the buffer, the following robot occupies the position of the preceding robot. The communication *pickup-to-robot* consequently uses information coming from robots to update the *open/closed* state, to request robots to enter the buffers and to

update the number of robots in queue. Besides, parcels also communicate with pick-up buffers to inform the buffers of their arrival and to transfer knowledge about the type of parcels incoming at these stations. Pick-up buffers use the information regarding the arrival of parcels to update their number of parcels and compare it to their maximum capacity. When the maximum capacity is reached, no other parcels can be held at the buffer. The type of parcel is another message exchanged with parcels that enables pick-up buffers to update their *open/closed* state, together with the number of hosted robots. In fact, parcels with higher load and volume require a higher number of robots to be loaded and transported. Therefore, when the incoming parcel is heavy and with high volume, the buffer will stay open until enough robots have entered the buffer to load and transport the parcel.
Parcels communicate with drop-off buffers in order to update the number of parcels collected into the containers. Drop-off buffers can contain a limited number of parcels, after which they are replaced with empty containers. In this model, however, we assume that drop-off buffers can contain an infinite number of parcels. This assumption does not have a strong impact on the operations of the robots, considering that it takes few seconds to a warehouse operator to replace a full container with an empty one.

Finally, the environment communicates unidirectional with parcels to stochastically assign them to pick-up buffers and assign destinations to parcels. As stated earlier, the environment only affects the states or behaviors of agents and objects, without being affected reciprocally by them. In the next paragraph, the process events are descripted to exhibit the actions of the agents and objects in this model.

4.3.4 Process Events

In the previous phase, we have identified the system elements, the environment in which these elements are placed and the interactions between system elements and between system elements and environment. However, the comprehension of how these system elements are supposed to behave in this industrial environment is still imprecise. For this reason, we start presenting a process flowchart (Figure 25) that straightens out our knowledge of the actions executed by robots in the environment.

The robot process flowchart embodies six main phases and the communication flow is also integrated in it. As stated earlier, robots communicate in their environment with other robots (robot-to-robot), parcels (robot-to-parcel) and pick-up buffers (robot-to-pickup). The six phases executed by each robot in the environment are namely the

“Search for the buffers”¸ i.e. map-building of environment, *“Approach pick-up buffer”, “Inspect the pick-up buffer”, “Inspect parcel”* at the pick-up buffers, *“Individual or cooperative pick-up”*, *“Individual or cooperative transport”.* These robot activities correspond to the main functional requirements for a multi-robot sorting system, e.g. move towards pick-up buffers to collect parcels; move with parcel towards destinations of parcels; release parcels; position in a queue behind the last robot etcetera.

Following, a description of these six phases (*functional requirements*) of the robot process is reported.

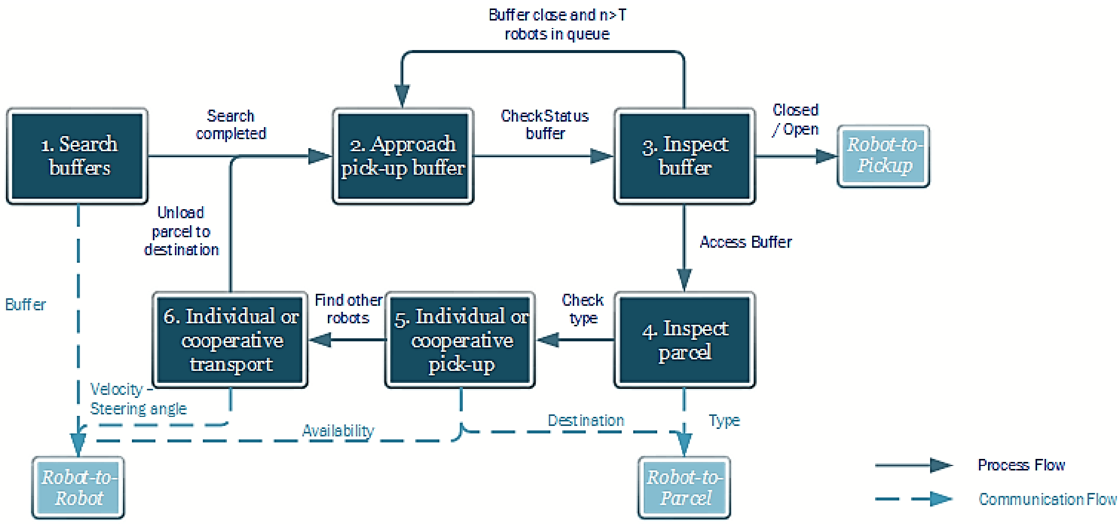


Figure 25: Robot Process Flowchart

Search for the buffers. At start, robots are placed in an unknown environment. Therefore, they need first to understand their surroundings by moving around, storing the positions of the pick-up and drop-off buffers and communicating the findings with other robots. This *“Search for the buffers”* phase is tackled in a distributed way. In fact, the area is divided into equally spaced parts and groups of robots are placed in each portion of the space. When a team completes the mapping of the assigned portion, it communicates the found buffers to the other teams (robot-to-robot). The communication robot-to-robot and the distributed approach enable accelerating the map-building process. The ability to reconstruct the map of the area provides the robots with the flexibility required to cope with changing environmental conditions, e.g. potential layout modifications. This also increases the reusability / adaptability of the system, which can be potentially employed in different settings (D. Fox et al., 2006).

Instead of using robots searching for buffers, thus constructing a map on their own, this map-building step could be easily executed by providing each robot with the map

of the environment in which they are placed. Nevertheless, the ability of robots to construct a map on their own allows operators to change the layout within a single shift, promising a higher degree of flexibility. In the event of layout changes, robots have to perform the map-building process another time. When the mapping is completed, robots transmit the map of the environment to a central computer. In case, additional robots are required during the shift, the central computer transfers the map to the extra mobile devices in order to avoid reconstructing the map a second time.
It must be noticed that in real-life, the map-building phase should be executed far up front, with the sorting operations starting only when robots have again full potential. This first phase does not affect the sorting operations; however, considering our aim to design a flexible and adaptable system, we suggest a method to perform this operation, without devoting excessive attention to it. Once the search is completed, the second procedure begins.

Approach pick-up buffer. At this point, robots are familiar with the environment in which they are placed. Subsequently, robots should be positioned to starting locations, from where they initiate their transport tasks. The starting locations are the queues of the pick-up buffers. Therefore, a proportionate number of robots is placed on each queue, from where robots enter their assigned pick-up buffers on a first-in queue first-served basis. After robots deliver their parcels to the right containers, robots need to re-calculate their pick-up buffers and move towards them. This corresponds to a resource allocation problem, where robots are the resources that require allocation. As we show later, this problem is solved through a heuristics algorithm, precisely a min-max heuristics algorithm. This algorithm enables balancing the number of robots over pick-up buffers, by partially striving to minimize the maximum waiting time of parcels at pick-up stations and partially making random decisions. In Chapter 3 (Table 4), we have showed that many other algorithms can be used for resource / task allocation problems, such as the Hungarian Method or auction-based algorithms. We detail later why we deem the min-max algorithm suitable for this application. We also explain why assigning each robot to a fixed location is not an adequate solution, given the effects of the environment (exogenous variables) on the destinations of parcels and consequently the different distances travelled by robots.
When a robot arrives at the first position in the queue, the next phase starts.

Inspect buffer. Once a robot arrives at the desired pick-up buffer, it communicates with it to understand its *open/closed* state (robot-to-pickup). Every buffer can be open or closed, meaning that it can host only a limited number of robots on it, based on the number and type of parcels that are contained in the buffer areas. For example, if there is only a low volume and light parcel in the buffer, the maximum number of robots that a buffer can host is one. Whereas, if there is a heavy parcel in the buffer, the maximum number of robots is equivalent to the minimum number of robots needed to transport that parcel.

Thus, in case of light parcels a buffer is open if:

$$\sum_i r_{il} < \sum_j p_{jl} \quad \forall\, l \in \{1, \dots, U\}, n \in R, k \in P$$

In case of high volume and heavy parcels, a buffer is open if:

$$\sum_i r_{il} \geq \sum_j p_{jl}\ q_{min} \quad \forall\, l \in \{1, \dots, U\}$$

$$where\ \ q_{min} = int\left(\frac{w_{p_j}}{f_{r_i}}\right), minimum\ number\ of\ robots\ required$$

$$with\ \ w_{p_j} = weight\ of\ a\ parcel\ and\ f_{r_i} = payload\ capacity\ of\ a\ robot$$

The above formulas stand for heavy or light parcels, yet they only take into account the weight factor. Nevertheless, parcels can be light and with high volume or heavy and with high volume. In this research, we only consider light - low volume and heavy - high volume parcels, since these type of parcels cover a great proportion of parcels. Furthermore, in consideration of the primary focus of this project, being the dynamic switch of behavior when facing ST-SR-IA (i.e. tasks that do not require cooperation of agents) and ST-MR-IA (i.e. tasks that require cooperation of agents) and the cooperative transport of parcels, these types of parcels are sufficient and relevant to demonstrate our hypotheses. Pertaining to light and low volume parcels, we assume that individual robots are sufficient to transport these loads.

Heavy and high volume parcels require the joint effort of more robots in order to be loaded and delivered. Here, the assumption we make is that four robots should be coordinated to load and transport these types of parcels. We believe this assumption is effective considering that robots should be small enough to allow leveraging hundreds of robots in a warehouse and increasing the amount of parcels delivered per unit of time. Furthermore, same as the wheels of vehicles, four robots can provide a good support for the loading and transport of large parcels.

Concluding a buffer is open when the number of robots is sufficient to tolerate the weight and size of the parcels contained in this area. When the parcel is light – low volume a single robot enters the area, while when the parcel is of high weight and volume four robots access it. If the buffer is closed, robots will queue up and wait in line before accessing this buffer area. However, if the number of robots waiting in line exceeds T (maximum number of robots that can wait in a queue), robots will move to another pick-up buffer. In this way, the distribution of robots over multiple pick-up buffers is increased. In the results, we show that thanks to the use of the min-max heuristics algorithm, robots waiting in line never exceeds *T*. When the buffer is open, the robot enters the buffer and the procedure four starts.

Inspect parcel. Once inside the buffer, robots communicate with parcels (robot-to-parcel) to reserve them and eliminate the risk of deadlocks or situations where multiple robots argue for the same parcel. Indeed, when a robot claims a parcel, the couple (r_i, p_j) is formed and other robots can no longer request the same parcel. This is a one-to-one assignment, where one robot is assigned to one parcel and vice versa. Robots change their load state into loaded, in order to prompt the subsequent actions. Furthermore, as described earlier, robots communicate with parcels to understand the type of parcels they need to transport.

Earlier, we have stated that heavy and high volume parcels cannot be loaded and delivered by individual robots. The one-to-one assignment, thus, might create some disorientation, since for these types of parcels the assignment may be considered a one-to-four assignment, where four robots load and transport these parcels (see 3rd phase to understand why four). Nevertheless, the assignment remains one-to-one, given that one robot is in charge of the task while the others are only responsible to help this robot accomplish its mission. This follows the leader-follower algorithm that we explain later in this Chapter, where only one robot is responsible for the task and control unidirectional the other robots to achieve it. After robots communicate with parcels and discern the type of parcel (task) they are assigned to the next phase starts.

Individual or cooperative pick-up. At this point, robots have recognized the type of parcel (task) they are assigned to. In this phase, robots will take the decision to either cooperate or act individually. If the parcel is small-sized and light, an individual robot is sufficient to load and transport a parcel, thus robots will act without help. This

implies that parcels are picked-up and transported to the right destination. In this model, robots execute directly the loading / picking operations.
In practice, warehouse operators or robot arms can be used to execute the loading operations, placing parcels on ground robots. Otherwise, UAVs can be deployed in place of ground robots to perform the picking and transport operations. In this dissertation, we use the term robot generically, without explicitly state whether robots in this application should be ground or aerial robots. Evidently, these two categories present very different characteristics and levels of complexity. However, in 2D, UAVs and ground robots can be seen as relatively similar entities. Therefore, without diving deeper into the difference of these classes, we assume the term robots and the entities in the simulation models analyzed can be observed as both UAVs and ground robots.

The picking operations require a temporal interval to place the parcels on the robots (or robots to pick up a parcel). According to data received from PostNL, loading operators can process 900 parcels per hour. Therefore, this temporal interval is assumed to be around four seconds, as this is the average time a human operator takes to place a parcel on traditional sorting systems. The choice of the destinations (containers) to follow depends on the *my destination* state of parcels. Parcels transmit this information via parcel-to-robot communication. Once acquired this information, robots activate their next action.

However, this only concerns small and light parcels. When an individual robot is not enough to load and transport a parcel, robots need to cooperate with other robots. The cooperation in this model follows a leader-follower algorithm. In Chapter 3, we have described this algorithm and observed how this was used in many scientific studies (J.P. Desai et al., 1998 and 2001; H. Sira-Ramirez and R. Castro-Linares, 2010; N. Noguchi et al. 2004; C. Zhang et al., 2016). Later in this Chapter, we describe how we have formalized this algorithm. In short, the robot detecting first the heavy and high volume parcel is assigned with the role of *leader* (or *master*). A leader communicates with other robots in the same pick-up buffer in search of potential followers. A robot can become a follower of a leader, as long as its load status is still set to unloaded, which means that it does not possess a parcel yet, and it is not a follower of another leader. Therefore, the leader can be seen as interpreting the role of an auctioneer, which searches and selects the most adequate robots to become its followers (combination *leader-follower* and *auction-based* algorithm). When the leader has recruited the minimum number of *followers* required to lift and transport the parcel, it creates a formation with the team and the parcel is loaded on them.

Concluding, when facing a complex task, robots can change dynamically their roles, which means that they can become either leaders or followers. By doing so, robots are able to cooperate to accomplish tasks, which are too complex for a single robot. After the parcel is picked-up, a leader communicates with its parcel and acquires the information related to its destination. Now, the team of robots is ready to transport the parcel to destination.

Individual or cooperative transport. Loaded robots, i.e. robots with parcels, are ready to transport parcels to destination. This corresponds to a path-planning and collision avoidance problem, where robots need to find the best path to arrive at destination without crashing with other robots. In Chapter 3, we have presented a list of algorithms that can be adopted to solve these problems. In this research, we use a combination of a shortest-path algorithm, similar to Basic Theta* or Phi*, and swarm intelligence behaviors, as proposed by D. Sun (2014).

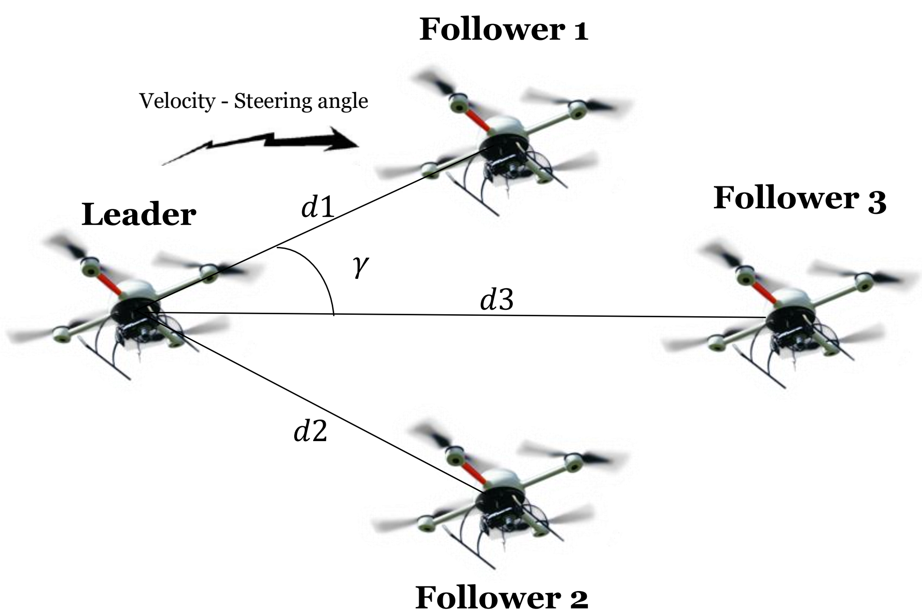


Figure 26: Leader-follower structure

The transport of light parcels is a more elementary task, with robots finding the shortest path to transport the parcel to the right destination and, while traveling through their selected path, avoiding collisions with other robots. Once at destination, robots unload their parcels. Same as loading operations, unloading operations require a temporal interval of few seconds to unload a parcel. Again, the temporal interval used in the model corresponds to the average time employed by human operators to unload a parcel from a traditional conveyor system. This time is of five seconds, thus higher than the time required to load parcels on robots considering that an operator has to remove the parcel from a robot and place it appropriately onto a container. At this point, the load status of robots is reset to unloaded and the robot is again available for another task.

The cooperative transport is a less elementary task, since the followers need to follow their leader with a given distance d_i and a relative angle γ (lateral and longitudinal offset). The leader controls unidirectional its followers, frequently sending them its velocity and steering angle. Figure 26 shows the formation of the team of cooperative robots with UAVs, one leader and three followers. The same formation is applicable to UGVs.

The leader coordinates the actions of multiple robots, deciding the path to follow and avoiding obstacles. Like the wagons of trains, followers convoy the leader with the objective of preserving the formation, with equal initial distance and angle. Moreover, followers receive from the leader information regarding the velocity and steering angle they need to possess. Thus, when the leader turns right, it will command the followers to turn right and so forth. The group moves preserving the formation until the destination is reached. Subsequently, the parcel is unloaded and the formation is dissolved.

The system is continuous without interruptions. Therefore, when the first parcel is taken to the appropriate drop-off buffer, the robot will go back to collect another parcel. This routine movement is executed until the completion of all parcels or the end of the shift time.

4.3.5 Algorithms to address the process events

In the previous paragraph, we have described the processes executed by robots within this application domain. It has been also anticipated the use of some algorithms to enable robots perform certain actions effectively. The described steps coincide with the domains of MRS, detailed in Chapter 3, where:

- *"Search for the Buffers"* is equivalent to a localization and mapping problem;
- *"Approach Pick-up Buffer"* is equivalent to a resource allocation problem;
- *"Individual or Cooperative Pick-up"* is equivalent to a task allocation and motion coordination problem;

- *“Individual or Cooperative Transport”* is equivalent to a path planning and obstacle avoidance problem.

4.3.5.1 Map building algorithm

In order to solve the localization and mapping problem, we use a multi-robot coverage coordination algorithm, where robots communicate their findings with their neighboring robots to guarantee a complete and non-overlapping partitioning of the area. Therefore, the environment is initially divided into identical partitions of space, and groups of robots are positioned in each partition. Like in D.L. Martínez and A. Halme (2016), robots are ordered in circle and drive forward without interfering with the trajectories of other robots. In each partition of the space, pick-up and/or drop-off buffers are placed. When robots sense a pick-up or drop-off buffer in the area, it saves the position of the buffer in memory if the buffer is not yet found and it immediately communicates the findings with the robots that are placed in the same area. Robots save the position of the buffer, only if this information is not already saved in the list.

```
Distributed search for buffers
for all r_i in R do
  list pickup = [number of pickup]
  drive forward with speed = max speed
  if u_l in radius = d and angle = θ then
    save u_l in temporary variable
    while i < list pickup length and found pickup = false and found
    empty = false do
      if list pickup [i] = u_l then
        found pickup = true
      else
        if list pickup [i] = empty then
          found empty = true
        else
          i =+ 1
        end if
      end if
    end while
    if found empty = true then
      list pickup [i] = temporary variable
    end if
  end if
end for
```

Figure 27: Map-building algorithm

When robots arrive at the end of the space, the area has been investigated and all buffers have been identified and correctly saved in memory. When every partition of the space is mapped, each group of robots transmits its information to the other groups, which save it into their list. In this way, the mapping problem is rapidly solved in a distributed manner.

Figure 27 shows the way we have conceptualized the algorithm for the distributed search for pick-up buffers. The same algorithm is used for the distributed search for drop-off buffers. For the whole algorithm, refer to Appendix A.

4.3.5.2 Resource allocation algorithm

The resource allocation problem, i.e. how to allocate robots to pick-up buffers to improve the utilization rate of pick-up buffers and of robots, is addressed using a min-max heuristics algorithm. A similar heuristics was proposed in A. Farinelli et al. (2017), where the objective was to maximize the throughput (i.e. number of tasks performed per unit of time) while minimizing the spatial interferences for robots (i.e. minimize the travel time for all robots). In their study, however, the task assignment corresponds to a ST-SR-IA, while in this application we are facing ST-SR-IA together with ST-MR-IA tasks. The differences between these two problem domains are noticeable, since an optimal distribution of robots in the former might be a poor quality solution in the latter. In fact, an equal distribution of robots over tasks fits sufficiently ST-SR-IA problems, but less ST-MR-IA tasks. In these problems, more robots should be assigned to ST-MR-IA tasks that require the cooperation of multiple robots, while ST-SR-IA tasks can be performed by individual robots. Therefore, the allocation of robots should be balanced in such a way that pick-up buffers with heavy and high volume parcels (ST-MR-IA) are assigned to a higher number of robots in comparison to pick-up buffers with light and low volume parcels (ST-SR-IA).

```
Allocation of robots to pick-up buffers
for all r_i in R do
  mypickup = nobody
  rand = extract randomly a number between 0 and 1
  if rand = 1 then
    save waiting time of parcels in list waiting time
    calculate max waiting time in list waiting time
    my pickup = pickup with max waiting time
  else
    save pickup buffers with distance < maxdistance in list potential
    pickup
    mypickup = one random pickup in list potential pickup
  end if
end for
```

Figure 28: Resource allocation algorithm

Furthermore, minimizing the distance travelled by robots in this multi-robot application would not produce optimal solutions, considering that the environment assigns the destinations of parcels randomly, there might be situations in which an excessive number of robots move towards the same pick-up buffer.

Having said this, in the heuristics algorithm we have developed, we exploit the information regarding the waiting time of parcels at stations, while keeping some randomness in the decision-making process of robots. Therefore, when deciding which pick-up buffer to follow, each robot either follows the station with the maximum waiting time in order to minimize it, or chooses one of the pick-up buffers within a certain distance from its relative position. When minimizing the maximum waiting time, robots are induced to follow the pick-up buffers containing heavy and high volume parcels. Indeed, ST-MR-IA requires more time to be executed in comparison to ST-SR-IA tasks, due to the number of robots required for their execution. Therefore, the waiting time for ST-MR-IA is generally higher in comparison to ST-SR-IA tasks. Moreover, when the decision is not to minimize the maximum waiting time, robots can decide to choose one of the pick-up buffers that are at a certain maximum distance from them, thus reducing the distance travelled and improving their overall distribution. We will show later the results of this algorithm, which are adequate for this application.

Figure 28 displays the way we have conceptualized the resource allocation algorithm used. The maximum distance depends on the relative position of the robots in the environment and the configuration of the warehouse adopted. In the next Chapter, we will explain how we have calculated the maximum distance for robots.

4.3.5.3 Task allocation & Motion Coordination Algorithm

In Chapter 3, we have described several algorithms that can be used to coordinate multiple robots in a formation. In this case, formations of robots are used to transport heavy and high volume parcels from pick-up buffers to containers. The algorithms that can be employed to build a formation and coordinate it include the leader-follower approach, the behavioral approach, the virtual structure approach, the graph-based approach and the artificial potential field approach. In the end of that paragraph, we have concluded that only the leader-follower and the virtual structure can be employed in this application, thus excluding the behavioral and artificial potential field algorithms. Graph-based algorithms can be used in combination with leader-follower and virtual structure algorithms.

Virtual structure algorithms considers the robots formation as a single virtual rigid body. This algorithm provides stability and ease to maintain the formation, considering that the system moves as only one rigid body. In the leader-follower algorithms, instead, one robot is considered the leader, while other robots are so-called followers, which have the task to maintain a desired distance and orientation to the leader and follow its trajectory. Both algorithms have the advantages of providing stability to the formation; however, both approaches have a centralized nature, which increases the fault-tolerance of the system. The virtual structure strategies depend on one centralized commander, which defines the formation to build and the trajectory to follow. The member of the formation communicates continuously to prevent a member to leave the formation. The leader-follower strategies depend on the leader vehicle to achieve its goal, thus the leader becomes the central system for a formation.

In this application, we have opted for a leader-follower algorithm. The leader-follower algorithm enables a substantial reduction of communication, since only one leader transmits commands to the other members. In many robotics applications that involve motion coordination of multiple devices, the leader-follower algorithms were adopted in order to reduce the communication between robots (M. El-Zaher et al., 2012). Furthermore, the leader-follower algorithm reduces the calculation required for path planning and collision avoidance. In fact, the leader only computes the best path for the formation and detects obstacles during the transport of parcels, with followers maintaining the same lateral and longitudinal offset from the leader. This provides stability to the formation, given that the followers do not calculate another path or avoid obstacles in a different direction than the one of the master. Furthermore, as explained in L. Consolini et al. (2008), the leader-follower provides higher scalability compared to the virtual structure approach. Introducing additional robots in a virtual structure composition affects the physics of the rigid body, decreasing the scalability of the algorithm.

In addition, we combine the leader-follower approach with an auction-like algorithm for the recruitment process of followers. J. Guerrero and G. Oliver (2012) propose a similar solution to address ST-MR-IA tasks, with a leader auctioning to form coalitions. When a robot becomes leader inside a pick-up buffer, it decides which other robots become its followers. In this case, the leader chooses its followers based on whether they are not already assigned to a task, they are not followers of other leaders and they are assigned to the same pick-up buffers. In this auction, only the leader decides the most appropriate robots for the task. Adding to this work, we have implemented a solution for the dynamic switch of roles for robots to address both ST-SR-IA and ST-MR-IA tasks, within the same application. When a robot discovers a ST-SR-IA task, it decides to act in a non-cooperative manner, transporting the parcels individually. While, when a robot discovers a ST-MR-IA task, it becomes a leader and starts recruiting followers to operate cooperative transport of parcels.

```
Reactive algorithm for execution of ST-SR-IA and ST-MR-IA tasks
for all r_i in R with load state = unloaded do
  leader = nobody
  load state = unloaded
  n followers = 0
  min robots = 3
  follower list = [min robots]
  if my parcel p_j = heavy and large then
    ;; cooperative behaviour
    leader = myself
    load state = loaded
    find r_i with load state = unloaded and my leader = nobody
    save r_i in temporary variable
    while nfollowers < min robots and found follower = false and found
    empty = false do
      if follower list [n followers] = r_i then
        found follower = true
      else
        if follower list [n followers] = empty then
          found empty = true
        else
          nfollowers =+ 1
        end if
      end if
    end while
    if found empty = true then
      follower list [n followers] = temporary variable
    end if
    for all followers in follower list do
      save my leader = myself
      save load state = loaded
    end for
  else
    ;; non-cooperative behaviour
    save load state = loaded
  end if
end for
for all r_i in R with load state = loaded do
  get my destination of my parcel
  save destination = destination of my parcel
  if my parcel = heavy and large and destination = o_m leftside or middleside
  then
    formation pattern1
  else
    formation pattern2
  end if
end for
```

```
Formation Pattern
if destination = o_m leftside or middleside then
  ;; formation pattern 1
  To follower 1:
  save steering angle = my steering angle
  save lat position = my lat position + l
  save long position = my long position
  To follower 2:
  save steering angle = my steering angle
  save lat position = my lat position
  save long position = my long position - l
  To follower 3:
  save steering angle = my steering angle
  save lat position = my lat position - l
  save long position = my long position - l
else
  ;; formation pattern 2
  To follower 1:
  save steering angle = my steering angle
  save lat position = my lat position - l
  save long position = my long position
  To follower 2:
  save steering angle = my steering angle
  save lat position = my lat position
  save long position = my long position + l
  To follower 3:
  save steering angle = my steering angle
  save lat position = my lat position - l
  save long position = my long position + l
end if
```

Figure 29: Motion Coordination algorithm

Furthermore, we consider here two formation patterns, where the leader is positioned in such a way that its situational awareness is increased, to avoid potential collisions during the motion of the formation (see Figure 29). To further increase situational awareness and avoid potential collisions, the leader of a formation uses sensing data deriving from its neighbors. In J.P. Desai et al. (2001), we have seen that formation patterns can be changed using graph-theory approaches in combination to leader-follower algorithms. In their research, the formation changed shape in accordance to the obstacle found on the trajectory. Evidently, in this domain, robots cannot change their formation when finding obstacles, considering that the main objective is to give stability to the parcel they are transporting. Nevertheless, multiple formations could be studied based on the shape of the parcels. For example, a rectangular parcel could be transported by a rectangular-shaped formation, whilst a triangular parcel could be transported by a triangular-shaped formation. In this research, we only consider square parcels; therefore, the formations have a square-like shape. The difference in the formation patterns is only the position of the leader. For the entire algorithm, refer to Appendix D.

4.3.5.4 Path Planning & Collision Avoidance

In Chapter 3, several algorithms for path planning and collision avoidance were investigated. D. Sun et al. (2014) propose a method to tackle this problem, by combining traditional path planning algorithms with swarm intelligence techniques. This behavior-based multi-robot path planning and collision method is proven adequate to coordinate a large number of robots performing transportation tasks in crowded indoor environments. Furthermore, it provides the benefits of good quality solutions, in terms of task completion times, great scalability and less computational time to calculate paths in comparison to other methods. In this research, we decide to use a similar algorithm, but instead of using A* for computing paths we opt for an algorithm that allows robots searching in every angle, e.g. Basic Theta* or Phi* (F. Duchon, 2014). This allows us to be less dependent on grid-like structures, used by all the MRS analyzed in Chapter 2. The elimination of this dependency increases the adaptability of our MRS, considering that there is no longer need to neither build a metallic grid-shaped frame, like in the Ocado or AutoStore systems, nor to use bar-coded stickers in a grid, like in the Kiva or in the latest STO Express multi-robot sorting system.

Using this algorithm, first every robot calculates its shortest path to destination, without considering the paths of the other agents. During the execution of the transportation task, each robot selects the most appropriate behavior to avoid or dodge other robots by temporarily deviating from their optimal path. In this application, several traffic situations can lead to a collision between robots. Robots need to handle potential frontal collisions, in particular between loaded robots that transport a parcel and unloaded robots that move towards pick-up buffers to receive a parcel. Furthermore, robots driving in different directions can originate side collisions. Finally, other collisions can occur during the queueing process and when releasing a parcel at drop-off buffers (see Appendix B for queuing algorithm). When facing head-on collisions, two robots activate a safety distance to force apart from each other. Therefore, a negative force of acceleration is given that leads the robot to arrest its motion. Then, both robots turn right, dodging a frontal collision.

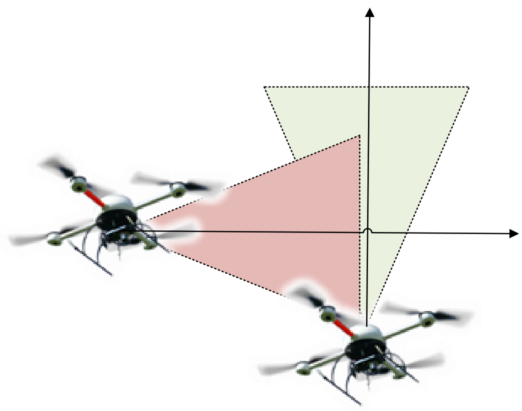


Figure 30: Example of side-collision avoidance

When the frontal collision is avoided properly, which entails that the other robot is no longer in sensory sight, robots accelerate again and move towards their destination. Side collisions are addressed by prompting a different behavior. If a side collision is forthcoming, one robot waits for the other to pass before moving forward. In Figure 30, a side-collision situation is shown, where a robot is going to the right and another robot drives straight. As can be seen in figure 30, the sensor of the robot going to the right has detected another robot. Therefore, this robot activates a waiting behavior to let the other robot go through the intersection and then move again. Other collisions can occur during the queueing process or the parcel unloading phase. A robot that enters a queue detects the last robot in the queue and maintains a safety distance from that robot. When the robot in front moves forward, the precedent robot follows it with the same speed and orientation. During the unloading phase, one robot or a formation of robots can enter the drop-off buffer. When a robot is already docking to a drop-off station, the success robots activates a wait for docking behavior to wait for the first robot to complete its unloading operations.

In the next paragraph, we show that different collision avoidance strategies need to be adopted based on the different traffic design strategies implemented. Certain traffic design strategies require a priority mechanism to assign higher priority to loaded robots and lower to unloaded robots. Figure 31 exhibits the basic specifications of the developed multi-robot collision avoidance. For the whole conceptual algorithm used, refer to Appendix C. Remember that different collision avoidance strategies are implemented for different traffic design strategies, but they are all based on the combined behavior-based multi-robot path planning and collision method, described above.

```
Multi-robot collision avoidance
for all r_i in R do
  compute shortest-path to destination of my parcel
  drive forward with speed = v
  if r_i in radius = d and angle = θ then
    save steering angle of r_i
    if side collision then
      speed = −v
    else
      if steering angle = my steering angle +180 then
        turn right by a number of degrees
      end if
    end if
  else
    speed = v
  end if
end for
```

Figure 31: Multi-robot collision avoidance algorithm

4.3.5.5 Deadlock avoidance vs autonomous robots

Deadlocks are situations involving two or more robots that find themselves at an impasse, i.e. a point where no progress can be made. In this system, there are many inter-robot strategies that can cause deadlocks, such as task allocation (allocation of parcels to robots), cooperative transport, and path planning and collision avoidance. Conflicts over resources can generate deadlocks when for instance two robots compete for the same parcel, leading one robot in a standstill state. To avoid deadlocks, we have synchronized the parcel-to-robot assignment (1-to-1 assignment).

However, other deadlocks can be induced during the procedure to avoid collisions. As seen earlier, collisions are avoided by using stopping and resuming policies for side-collisions and right moving for frontal collisions. Moreover, we have calculated the coverage of the area per robot to guarantee that robots in the area have enough free space to move. In addition, the speed of robots is kept voluntarily low. In this way, we have been able to avoid most of the deadlock situations. Nevertheless, physical deadlocks may still occur, due to the autonomy of robots and, consequently, the lack of global knowledge. Deadlocks can be avoided if robots can predict and anticipate all the trajectories of other robots and, consequently, plan and re-plan collision free paths. Using our collision avoidance algorithm, robots plan independently their path following it and retouring only in certain instances when they meet other robots with conflicting trajectories. The high autonomy of robots provided by these strategies (lack of global knowledge, limited and local communication, decentralized collision avoidance algorithm) can generate physical deadlocks, in which robots mutually prevent the motion of other robots (*circular wait*). This typical deadlock situation is called deadlock cycle.

To resolve distributed deadlocks, robots must first understand the outbreak of these situations (*deadlock detection*). K.M. Chandy et al. (1983) suggest the use of graph theory, in which each robot represents a node of a graph. This algorithm requires each robot to know the robots that are impeding its movement (in graph theory these are the outgoing edges of a node). In this situation, one robot (origin node) sends a message along the cycle. When the second robot (node) receives the message, it forwards it to the next node. If the origin robot receives the same message it has initiated, the deadlock is recognized. Other deadlock detection algorithms exist that involve the use of Petri net, in which robots exchange tokens instead of messages (Y. Zhou et al., 2017). For the resolution of deadlock cycles, centralized and decentralized algorithms can be adopted. Centralized methods include using hierarchies to set ordering motions (A. Silberschatz et al., 1998) or assigning robots with different time delays (X. Wang et al., 2015) and re-planning the trajectories of robots. M. Kloetzer et al. (2013) suggest the use of Petri net for deadlock prevention in robot planning, setting restricted capacity to some regions of the environment, meaning that a limited number or robots can cover simultaneously the same area, and assigning robots with sets of possible trajectories. M. Jager and B. Nebel (2001) propose a decentralized algorithm that consists of two steps. First, one robot in a circle is sent a message to change direction to destroy the circle. However, other robots in the neighbors (not in the circle) might hinder this re-planning strategy of a robot. If this strategy is not successful, then the robots in the circle ask neighbors that are not in the circle but hinder their motion to plan alternative trajectories. Using this approach, space is created to robots to re-plan their trajectories. It is however important to avoid all robots to plan simultaneously alternative trajectories at the same time, but this number should be instead kept low to resolve deadlocks.

In our system, we propose two approaches that should be tested with simulations. When a deadlock cycle is detected, robots analyze their load status. Using this information, higher priority is assigned to loaded robots and only these robots re-plan their trajectories using a collision free path. In this way, the deadlock cycle is broken and the other robots are able to move again. Using this approach, the only variable to consider is the load status of robots. The use of priority schemes based on load status of robots could be used for collision avoidance, with robots recognizing their obstacles to a longer distance and deciding to stop and leave them enough space to move or continue their motion. However, using this strategy, higher knowledge of the states of other agents is required. The second approach consists in a random decision to assign priorities to robots. Knowing that robots are encapsulated entities with unique ID, we assign priorities in an ascending order. Hence, the robot with the highest ID number moves first, while the robot with the lowest number moves last. In this research, we do not test these strategies using simulation; therefore, other researchers can test these strategies to try to resolve deadlock cycles.

Another deadlock situation arises during the cooperative transport of robots. In this situation, when one robot stops working, the other robots in the formation can no longer move. In Chapter 5, we propose a solution to resolve these deadlocks, using a robot assistance mechanism involving other robots helping trapped robots move again.
Finally, it is necessary to underline that the higher the autonomy of robots the higher the probability deadlocks surface. Moreover, it is highly complex to solve all deadlocks deriving from robot motion planning and collision avoidance. It is therefore essential for robots to promptly understand the presence of deadlocks and solve them before this problem propagates to the N other robots in the system. Providing robots with higher knowledge might be required to solve all these situations.

4.4 Performance Indicators for parcel-sorting MRS

Following the Modeling Development Process of paragraph 4.2, the next step would be the development of computerized model through computer programming and implementation. Nevertheless, we find it appropriate to define first the inputs and outputs of the model before implementing it in a software.
Inputs are those parameters of the model that can be altered in order to evaluate the effects of every alteration on the outputs. Outputs correspond to the results of the simulation runs, which are used to assess the performance and adequacy of the model and compare it with real-life applications.
The parameters that can be changed in the simulation are the following:

- ***Number of robots***;
- ***Percentage light and low volume / heavy and high volume parcels***, e.g. 90 percent light and low volume parcels and 10 percent heavy and high volume parcels;
- ***Number of pick-up buffers***;

- ***Number of drop-off buffers***;
- ***Maximum pick-up buffers capacity***, i.e. maximum number of parcels per pick-up buffer;
- ***Speed of robots****,* i.e. speed of robots without parcels or carrying light and small loads;
- ***Speed of cooperative robots****,* i.e. speed of robots in a formation, carrying heavy and high volume loads;
- ***Position of drop-off and pick-up buffers****,* i.e. change placement of buffers;
- ***Temporal interval to unload/load parcels***, i.e. time waited by robots when obtaining or realising parcels;
- ***Arrival parcel distribution***, i.e. distributions of parcels across pick-up buffers (*uniform/non uniform*);
- ***Traffic design alternatives****,* i.e. whether robots with parcels share the same workspace with robots without parcels or these are separated within the same plane or different planes;
- ***Number of faulty robots****,* i.e. number of robots that are no longer able to carry out their assigned tasks;
- ***Number of faulty pick-up buffers****,* i.e. number of pick-up buffers that are no longer available for the assignment of parcels;
- ***With or without assistance mechanism****,* i.e. other robots placed outside the transport field can intervene every time robots fail to remove them from the transport area.

As already described in Chapter 1 paragraph 1.3, system effectiveness, congestion and fault tolerance represent the KPIs for this multi-robot system. System effectiveness is one of the most relevant KPIs for the design of a new system. In their research, A. Farinelli et al. (2017) assess the ability of a multi-robot system to maximize system effectiveness, in terms of objects transported per unit of time. Moreover, the authors also point out the importance of diminishing the number of robot interferences in order to minimize the average task completion time. Therefore, congestion is employed as another relevant performance indicator, which is tightly connected with system effectiveness. D. Sun et al. (2014) propose a multi-robot approach to minimize the completion times of transportation tasks. Thus, in their study, the authors only focus on system effectiveness. Z. Yan et al. (2013) underline the importance of analyzing system effectiveness of multi-robot systems, particularly by focusing on qualitative aspects like flexibility, scalability and versatility that provide these systems with potential superior performance. C.S. Kong et al. (2006) indicate congestion as a potential limiting factor for the effectiveness of multi-robot systems. Z. Yan et al. (2013) stress on the importance of devising adequate coordination strategies in multi-robot environments to reduce congestion and increase system safety. L.E. Parker (1995) advocates the inadequate focus of previous work on the issue of fault tolerance, which according to the author, represents a key design issue for real-world multi-robot applications. L. Vig and J.A. Adams (2006) identify fault tolerance as a cardinal issue in multi-robot coalition formation. In their study on multi-robot patrolling, D. Portugal and R. P. Rocha (2013) discuss the negative influence of centralized strategies and global knowledge on fault tolerance. We can therefore conclude that system effectiveness, congestion and fault tolerance are essential factors to examine for the design of a multi-robot parcel sorting system.

The conceptual MRS parcel-sorting system resembles to some extent a queuing system, where customers are parcels, queues are pick-up stations and servers are robots. For this reason, we can use some performance indicators that are typically employed to examine queuing models, such as utilization rate of servers, waiting time of customers and service time of customers (N. Kheir, 1995). Here, we need to quantify system effectiveness, congestion and fault tolerance using specific multi-robot performance indicators.
Therefore, the performance indicators used to quantify system effectiveness, congestion and fault tolerance are the following:

- ***Average Utilization rate of robots***: average number of robots with a loaded status. This indicator was used in C.J. Hazard et al. (2006), J.L.G. Sanchez et al. (2002), K.A. D’Souza et al. (1994) to measure the performance of robotic systems.

$$Average\ utilization\ Rate\ robots = \frac{\sum_i r\ load_i}{R} \quad with\ r\ load = robots\ with\ parcels \qquad [\%]$$

- ***Throughput***: number of parcels on drop-off buffers. This performance measure was employed in A. Farinelli et al. (2017), C.J. Hazard (2006), D. Sun (2014), J.L.G. Sanchez et al. (2002) to evaluate the accuracy of their multi-robot algorithms.

$$Throughput = \sum_m p_m \qquad [\#]$$

- ***Average Robot Performance***: average number of parcels transported by each robot. This performance indicator was used in K. Shirase and S. Aoyagi (2009) to measure the efficiency of multiple service robots.

$$Average\ performance\ of\ robots = \frac{\sum_m p_m}{R} \qquad [\#]$$

- ***Utilization rate of pick-up buffers***: average number of robots per pick-up buffer, meaning number of robots in the queue connected to a pick-up buffer plus the number of robots inside this buffer. This performance indicator was used in C.J. Hazard et al. (2006) and K.A. D'Souza et al. (1994).

$$Utilization\ Rate\ pickup\ buffers = \sum_i r_{il} \quad \forall\, l \in \{1 \dots U\} \qquad [\%]$$

- ***Average waiting time of parcels*** before being picked-up: average time waited on a pick-up buffer. This indicator was used in C.J. Hazard et al. (2006) and in C. Sung et al. (2013) to measure the correctedness of the developed task assignment algorithms.

$$Average\ waiting\ Time\ of\ a\ parcel = \frac{\sum_l \sum_j s_{lj}}{\sum_l p_l} \qquad [s]$$

- ***Average service time of parcels***: time a robot needs to take a parcel to the appropriate destination. We do not consider important to calculate the return time of robots, which in this system would not give extra information compared to the average service time of parcels. This performance indicator was used in A. Farinelli et al. (2017) and D. Sun (2014).

$$Average\ service\ Time\ for\ a\ parcel = \frac{\sum_m \sum_j s'_{mj}}{\sum_m p_m} \qquad [s]$$

- ***Total time in system for a parcel**:* it can be simply computed as the sum of the average waiting time and the average service time.

$$Tot\ time\ in\ system = Average\ Waiting\ Time + Average\ Service\ Time \qquad [s]$$

- ***Average robot distance travelled idle**:* calculated as distance travelled by robots without parcels (idle) divided by total distance travelled by robots. This performance indicator was used in C.J. Hazard et al. (2006), M. Elango et al. (2011) and R. Simmons et al. (2000).

$$Distance\ travelled\ by\ one\ robot\ idle = \frac{dist_{r\ load}}{dist_{r\ load} + dist_{r\ unload}} \qquad \forall\, i \in \{1 \dots R\} \qquad [\%]$$

$$Average\ distance\ travelled\ idle\ by\ robots = \frac{\sum_i^R distance\ travelled\ by\ robots}{R} \qquad [\%]$$

- ***Average Congestion***: measured as average speed of robots divided by maximum speed. This performance indicator was employed in A. Farinelli et al. (2017), C.S. Kong et al. (2006) and A.S. Tanenbaum (2003) for the analysis of the area coverage of robots and number of interferences among them.

$$Max\ robot\ speed = \max v_i \qquad [m/s]$$

$$Average\ robot\ speed = v_i \qquad \forall\, i \in \{1 \dots R\} \qquad [m/s]$$

$$Congestion = \frac{\sum_i {}^{v_i}/_{\max v_i}}{R} \qquad [\%]$$

- ***Conditional congestion***: time spent avoiding collisions, i.e. when speed of robots is lower than their maximum speed. This measure is recommeded by A.S. Tanenbaum (2003) for the analysis of the performance of multiple electronic devices.

$$Average\ collision\ time = \frac{\sum_i s_i|_{v<v\,max}}{n\ collision} \quad \forall\, i\ \in\ \{1 \dots R\} \qquad [s]$$

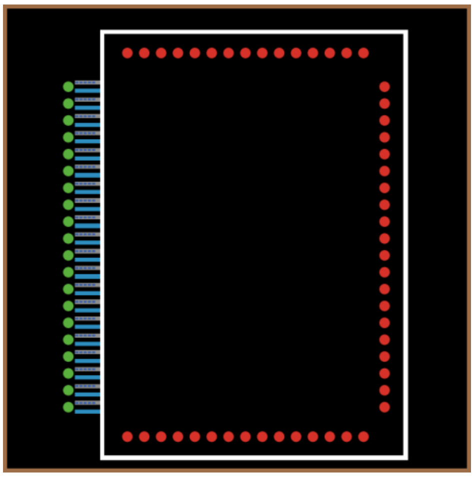
$$Conditional\ congestion = \frac{\sum_i^R ACT_i}{R} \cdot average\ n\ collisions\ avoided \qquad [s]$$

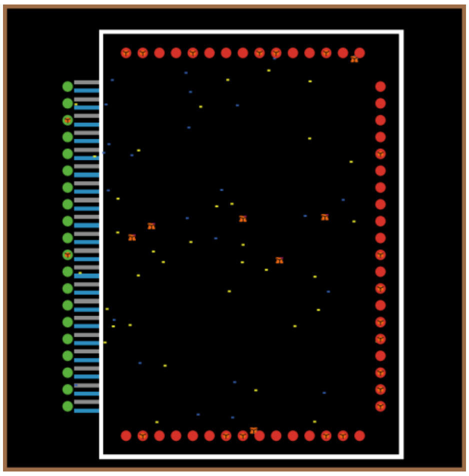
4.5 Software Implementation

Once the conceptual model is developed, the next step is to implement it in an appropriate modeling environment. In paragraph 4.1, we have anticipated the use of ABS for the implementation of the conceptual model. As already argued, ABS is adequate to implement models that relate to technical domains other than social sciences (P. Davidsson, 2000), given that:

- It enables to add or remove easily agents during the simulation without interruption;
- It facilitates distributed computations, thus allowing modeling large-scale technical systems;
- It supports structure-preserving modeling of simulated reality, meaning that real-life entities are well descripted in the simulation.

There are many available ABS tools for the implementation of models. We argue that NetLogo is a valid software for the scope of this research, without inquiring the differences between the various tools, describing the strengths and weaknesses of each tool. NetLogo is an open-source java-built ABS software and it is especially suitable when it comes to developing rapid prototyping, given its simple programming syntax. Furthermore, it possesses a vast online community and a large number of example models that can support systems engineers during the implementation of the models. Considering the large proportion of this system, where the actions of thousands of agents and objects are taken into consideration, NetLogo guarantees comparable easier implementation, scalability, easier modification of simulation parameters and accurate description of agent/object classes. Moreover, using NetLogo we are able to record conveniently a number of statistics about each simulation and draw graphs of the results. In NetLogo, the world is divided into patches, i.e. points of space where agents and objects are positioned. At beginning, the world is unbounded, with a shape of a donut/torus where turtles appear and disappear on opposite edges. However, settings are changed in order to create a bounded world, representative of a sorting center.





(a) (b)

Figures 32a-32b: NetLogo environment (a) and agents moving in the environment (b)

Taking inspiration from the Alphabet Soup model (see Chapter 3) and considering the layout of traditional sorting centers (see Chapter 2), we design the system displayed in Figure 32a. In this sorting environment, the entry gates, where the inbound ULDs arrive, are located on the left side. Instead, the exit gates, where the outbound ULDs are moved onto outbound trucks, are located on the bottom, top, and right sides. Correspondingly, the pick-up buffers are located on the left side (see green circles) and the drop-off buffers on the other sides of the environment (see red circles). In this environment, there are 20 pick-up buffers and 50 drop-off buffers. As can be observed, each pick-up buffer is connected to the transport field with two queues, one entry queue and one exit queue. Entry queues (grey-colored) are the places where robots wait to enter a pick-up buffer, while exit queues (sky-colored) correspond to the lines robots drive through before entering the transport field. The transport field (marked in white) signals the area where robots transport parcels to the appropriate drop-off buffers. The dimensions of this area are 75x50 patches (length + width). Assuming a patch is equivalent to one meter, this entails that the area considered is of 3750 m^2.
Furthermore, we have designed buffers with a diameter of 250 cm and robots with a size of 90x60 cm. The tiny entities that can be viewed in Figure 32b represent robots moving inside the simulated world. As stated earlier, robots have different load status, depending on whether they are transporting parcels or moving back to the pick-up buffers to collect other parcels. To make this difference noticeable in the simulation, we have marked robots without parcels with a blue color and robots with parcels with

other colors. Yellow robots are transporting light and low volume parcels, while orange/magenta are robots transporting heavy and high volume parcels. Furthermore, earlier we have underlined that every formation of robots is composed of three followers and one leader. Followers are assigned with an orange color, whereas leaders are assigned with a magenta color. In this situation, both robots with and without parcels share the same area, with the former moving towards the destinations of the parcels and the latter moving towards one of the pick-up buffers. In the next Chapter, we show that the traffic can be controlled differently, with loaded robots separated from unloaded robots on the same plane or on a different plane.

Time is another important aspect of the simulation that needs careful consideration. In comparison to reality, where time is continuous, our simulations run at discrete time steps, with ticks representing the smallest unit of time. Therefore, we have to define how much time a tick is meant to represent in our simulation. In this simulation, we assume that four ticks represent one second, giving a temporal precision of 250 milliseconds. Using a 4:1 ratio, robots travelling in real-time at 2 meters per second, have in the simulation a speed of 0.5 ticks per second. This entails a spatial precision of 500 millimeters. These temporal and spatial precisions are adequate; however, more accurate temporal and spatial precisions can be obtained by changing the ratio ticks/seconds, however higher temporal precision also entails lower simulations.

In the next paragraph, we discuss the model verification phase.

4.6 Model Verification

Model verification ensures that the conceptual model is correctly translated into the computerized model. This phase is supposed to build confidence into the implemented model by controlling that the computer code reproduces the specifications defined in the model conceptualization phase. Indirectly, model verification also allows detecting errors in the conceptual model. Frequently, it results problematic uncovering errors in the specifications of the conceptual model until the model is implemented and verified. In his book, I. Nikolic et al. (2013) proposes four strategies to verifying models:

- *Recording and Tracking Agent Behavior (RTAB)*, in which relevant output variables are selected and recorded.
- *Single-agent testing (SAT)*, in which the behavior of a single agent is verified.
- *Interaction testing (IT)* limited to minimal model, in which the interaction between agents is tested.

- *Multi-agent testing (MAT)*, in which the system behavior (behavior of multiple agents) is examined.

Table 10. Examples of verification tests

Verification type	Agent	Description	Expected Result	Obtained Result	Verified?
RTAB	Pick-up buffer	Controlling the Open/Closed state of pick-up buffers	Two binary variables switch to 0 when a robot cannot enter a pick up buffer and to 1 when the robot can enter the pick-up buffer	Correct update of binary values	V
SAT	Robot	Testing the behavior of robots in queue	Robots should position themselves behind last robot in queue and move to the next position every time a robot enters the pick-up buffer	The behavior of 10 randomly selected robots was observed. All robots update correctly their position in queue.	V
IT	Robot-to-parcel	Testing the 1-to-1 robot-to-parcel assignment	Parcels should pair up with the first robots in queue. Parcel set the variable *myrobot* to the robot to which it is assigned. This parameter does not change during the whole process.	The behavior of 10 parcels (of different types) is observed. Parcels correctly pair up robots that are positioned first in the queue. The variable *myrobot* updates to that robot and it does not change its value.	V
MAT	Robot-to-robot	Testing the collision avoidance with increased levels of speed	Increasing the speed might cause more collision to occur. We want to evaluate at which speed the robots start colliding. In this test, the number of robots is equal to 100. I expect that there are no problems with increasing the speed. Changing speed parameters: Speed = 2 m/s Speed = 3 m/s Speed = 4 m/s	Tests are carried out for 5000 ticks with 100 robots and 95% light and low volume parcels and 5% heavy and high volume parcels. No collisions are again noticed. Therefore, we can conclude that speed does not create problems to collision avoidance. Other tests should evaluate the impact on collision avoidance of increased number of robots.	V

Model verification experiments are conducted using the suggested strategies. Table 6 features the procedures we have used to execute the verification tests for each of the listed strategies. As can be viewed, when performing a verification test, we start identifying the agent(s) or the interaction agent-to-agent or agent-to-object to observe; next, a brief description of the desired test is reported; following, our expectations from the test are disclosed; finally, the results are obtained by implementing small tests manually or, directly, in the software platform. Many other model verification experiments can be found in Appendix G.

It is important to underline that model verification is never complete in all respects, considering that an endless number of experiments or combinations of experiments can be implemented. Furthermore, it is impossible to guarantee that a certain number of tests is enough. To build a good level of confidence into a model, it is important to perform verification experiments throughout the whole development of the model and not just during this phase.

4.7 Model Validation

Model validation confirms if the outcome of the implemented model has sufficient accuracy to achieve the intended purpose of this research. The Modeling Development flowchart (par 4.2) shows that model validation is carried out after the implementation of the computerized model and when opportune verification experiments are executed. Furthermore, in this comprehensive flowchart, we have indicated the relation between model validation and experimentation. Model experimentation, i.e. activity where inferences about problem entity are obtained by running simulations with changing input variables, is generally performed ex-ante or ex-post model validation.

In this project, we have opted for executing model experimentation ex-post to obtain, during model validation, insights into the experiments to implement in order to address purposefully the investigated research questions.

I. Nikolic et al. (2013) describes several methods to validate a model, namely:

- *Historic replay*, in which a model is compared to real-world situation;
- *Expert consultation*, in which structured interviews or workshops with experts are performed;
- *Literature validation*, in which the academic literature (theories, published case studies, etc.) is investigated;
- *Model replication*, in which a second model is created with different modeling choices.

In consideration of the problem entity for this research, the only two validation techniques suitable from the four listed above are *expert consultation* and *model replication*. In fact, comparing the simulated system with real-world situations is unfeasible, given that, in our knowledge, there exists only one similar system (see Chapter 2), but there is no data available regarding this system. Furthermore, *literature validation* is also impractical, since there are no published studies in the literature that we can use to validate our model. *Model replication* is a suitable option that can provide strong understanding over the validity of this model. Nevertheless, time constraints prevent us from replicating the model adopting different strategies. Therefore, the only remaining option between the suggested techniques for the validation of this model is *expert consultation*. The limitations related to this technique are reported in Chapter 6.

Two experts have been consulted for the validation of this model. The field of expertise of these experts fit adequately with the scope of this validation phase. The first interviewed expert is a full professor at the Technical University of Delft, in the transportation department of the faculty of Technology and Management. This professor has an extensive knowledge in freight and logistics and his areas of interest also concern agent technology, process optimization, collective behaviors and self-organizing systems. The second interviewed expert is a former Technical University of Delft student with a doctoral degree in transport and logistics. This expert masters proficiently simulation tools and has past experience in parcel sorting centers.
The implemented model was presented to both experts, after which a short interview was conducted. Both experts were asked the following questions:

i. *To what extent does the model represent real-life sorting operations?*
ii. *Considering the objectives of this research, do you believe that the model's outcome can provide answers to the investigated research questions?*
    - *Are the employed performance indicators adequate to make comparisons between experiments?*
    - *Can the model show if the MRS is well tolerant to failure?*
    - *Can the model show if the MRS is flexible?*
    - *Can the model show if the MRS is performing well, in terms of system effectiveness and not cost?*
iii. *What experimental designs do you think should be developed in order to fully answer the main research question?*
iv. *Any other remarks/considerations you want to make*

The experts recognized the model as a valid representation of real-life sorting operations, stressing how these operations may be adapted to the new system, given the level of flexibility provided by it. The experts recognized the potentiality of the system in terms of high tolerance and flexibility, and emphasized that energy consumption, maintenance and cost may be limiting factors for this system. In fact, the identified performance indicators are adequate and suitable to make comparisons between experimental designs, but other indicators should be included which constitute essential outputs for these systems. These additional indicators include:

- *Maintenance indicators*, which are crucial to maintenance planners to make optimal scheduling of activities;

- *Energy consumption indicators*, considering that each robot needs to be recharged periodically. When the number of robots is elevated, this factor can cause serious disruption;
- *Cost-related indicators*, which are the primary dimension of performance in logistics problems.

Furthermore, the experts indicate that, in real-life sorting situations, parcels are not distributed evenly among the pick-up buffers. Therefore, improvements should be made in the model, where pick-up stations have unequal number of parcels to dispatch. In addition, parcels may be assigned with higher and lower priority, indicating whether robots should prefer picking up certain parcels rather than others.
The experts emphasized the importance of developing experimental designs where the number of robots is varied, the percentage of light and low volume / heavy and high volume is altered, and experiments are implemented that concern with disruption to assess the robustness of the designed system. When these experimental designs are made and accurate graphs are drawn, the model's outcome can provide answers to the investigated research questions.

It is important to remark that expert validation is a valuable process to provide us (the modelers) with meaningful insight into which aspects of the model are satisfying and which deserve further attention. However, we need to focus on the aspects that are relevant for the scope of the research. For instance, energy consumption, maintenance and cost factors are crucial for a faithful representation of reality, but they are not equally important for the aim of this project. The goal of this research concerns with the development of a high-level representation of a MRS and with the evaluation of the impact of cooperative behaviors on the effectiveness, congestion and robustness of the system.

Concluding, both interviewees have highlighted the detailed degree of the developed model, underlying the potentiality but also the limitations that could negatively affect the performance of these systems. For the entire interviews, refer to Appendices H and I.

Conclusions

In this Chapter, we have first collocated our model into the broad spectrum of agent-based computing. This model corresponds to a multi-agent system (MAS); however, we have employed an agent-based simulation (ABS) tool for the implementation of this model. Furthermore, using the Modeling Development flowchart, we have described the main phases to follow when modeling a system, namely explore, model, test and evaluate. We have divided the model conceptualization phase into four steps, namely identification of system elements, definition of interactions among system elements, definition of process events and elaboration of algorithms to address the described process events. The main system elements to include in the model are robots, parcels, pick-up buffers and drop-off buffers. All system elements communicate in order to prompt rules and change state. The main process activities executed by robots are *search for buffers, approach pick-up buffers, inspect buffers, inspect parcels, individual or cooperative pick-up,* and *individual or cooperative transport.* These activities match with the challenging domains of MRS identified in Chapter 3, respectively *localization and mapping, resource allocation, motion coordination* and *path planning and obstacle avoidance*. These problems are solved using the following algorithms: *multi-robot coverage coordination; min-max heuristics; leader-follower; Phi* or Basic Theta** (NetLogo built-in) and *Swarm intelligence techniques*.

Table 11. List of assumptions

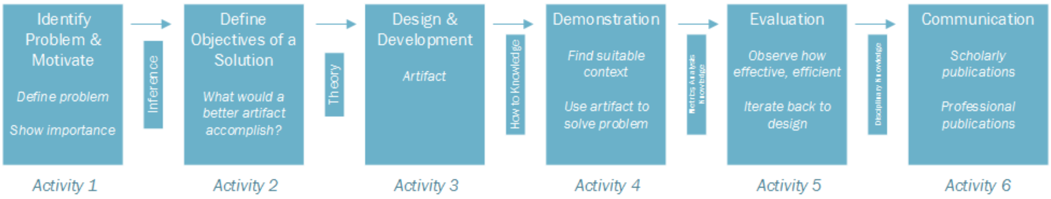
Assumption 1	Parcels arrive continuously at pick-up stations with equal distribution
Assumption 2	Parcels can only be of two types: light-low volume or heavy-high volume
Assumption 3	Robots with light-low volume parcels or empty travel at 2 m/s. Robots with heavy-high volume parcels travel at 1 m/s
Assumption 4	Robots wait 4 s at pick-up buffers and 5 s at drop-off buffers before leaving
Assumption 5	Warehouse has a rectangular layout and it has a total size of 3750 m^2
Assumption 6	In the sorting center, there are 20 pick-up buffers and 50 drop-off buffers
Assumption 7	Four robots are required to load and transport heavy and high volume parcels
Assumption 8	Robots are 90x60 cm and Buffers have circle shape and diameter of 250 cm

The performance indicators required to quantify system effectiveness, congestion and fault tolerance (KPIs) are: *average utilization rate of robots; average robot performance, throughput; utilization rate of pick-up buffers; average waiting time / service time of parcels; average robot distance travelled idle; average congestion; conditional congestion.* Model verification has been executed using the four strategies suggested by I. Nikolic et al (2013). Two experts of logistics and simulation have been consulted to carry out the model validation phase. Finally, we want to underline the assumptions described in this Chapter that are of relevance for the prosecution of the next phase, being *implementation of experimental designs*.

In Table 11, we present the main assumptions made in this Chapter. In the next Chapter, we describe and implement several scenarios and analyze the outcome of these experiments. The aim is to create an instance for further studies on the topic.

In Appendix F, we explicate the main design choices made in this Chapter, providing a retrospective on our design project.

5 | Experimentation



Introduction

In Chapter 4, we have built a conceptual model of a parcel sorting MRS and implemented it in a software typically used for agent-based models, termed NetLogo. Subsequently, we have verified and validated the model to build confidence into the results of simulations. This Chapter represents the fifth activity of the design methodology used in this dissertation, being *Evaluation*. In this phase, computer experiments are conducted by altering the values of certain input parameters and inferences are deduced, which intend to give responses to the research questions.

Before developing experimental designs, we describe three traffic design alternatives that are used to control differently the traffic flow of robots inside the sorting hub. References to the conventional traffic control of vehicles are made to build the different traffic control strategies. Subsequently, we detail our choices with regard to the input parameters to keep constant and those that we need to alter. By doing so, we are able to build insightful experimental designs, reducing the endless number of potential combinations of values. Guided by the objectives of this thesis dissertation, we develop the experimental designs and encapsulate them in a comprehensive table. Following, the results from the various experimental designs are reported in different paragraphs according to the pursued objectives, and these are analyzed thoroughly. After the end of this analysis, we compare the newly designed parcel-sorting system with the traditional sorters, also referring to Chapter 2. Eventually, in the conclusion paragraph, we combine all pieces of the puzzle, detailing out the main conclusions of this Chapter.

Therefore, the Chapter is structured as follows. In Paragraph 5.1, the three traffic design alternatives are presented and described separately. In Paragraph 5.2, the experimental designs are formed by making decisions regarding the parameters that are important to alter in order to obtain insights into the objectives this research. These experimental designs are then encapsulated in a table. In Paragraph 5.3, the results from the experiments are reported based on the objectives they pursue. In fact, in sub-paragraph 5.3.1, results concerning with the impact of cooperative transport on system effectiveness are described. In sub-paragraph 5.3.2, results concern with the impact of cooperative transport on congestion. In sub-paragraph 5.3.3, results concern with the impact of cooperative transport on system fault tolerance. In Paragraph 5.4, comparisons between traditional sorting devices and the new multi-robot parcel sorting system are made, also looking at data provided by PostNL. Finally, in the Conclusions Paragraph, we collect all the inferences deduced in this Chapter.

5.1 Traffic Design Alternatives

In Chapter 4, we have described the input parameters that can be altered in order to obtain different results from the simulation runs. Traffic design alternatives represent one of the input parameters that can be varied to manage diversely the traffic flow inside the sorting hub. In transportation, the traffic flow is managed using three basic approaches (T.R. Neuman, 1985):

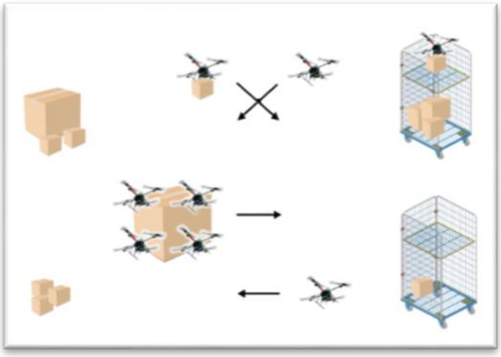
- Vehicles moving in opposite directions, on the same plane, share the same roads, turning right or left at intersections, e.g. in (most of) urban roads;
- Vehicles moving in opposite directions on the same plane do not share the same roads, e.g. in highways;
- Vehicles move in opposite directions on different planes, e.g. on elevated roads.

At the same way, robots can share the same field, moving on different paths or on different planes. In this layout (see Chapter 4.5), where pick-up buffers are located on the left sides and drop-off buffers on the other sides, loaded and unloaded robots move in different directions. For this reason, we want to analyze the differences between traffic configurations where robots share the same transport field or where robots are separated on the same plane and on different planes.

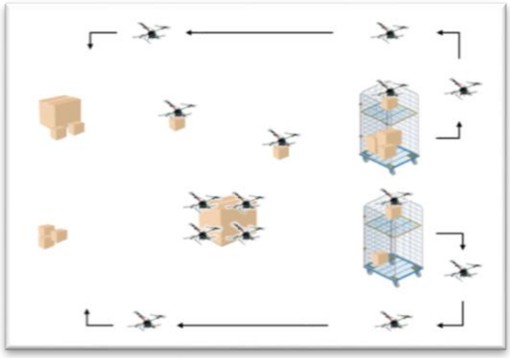
Accordingly, we develop three different traffic design alternatives:

- Mixed traffic, in which robots share the same field (Figure 33a);

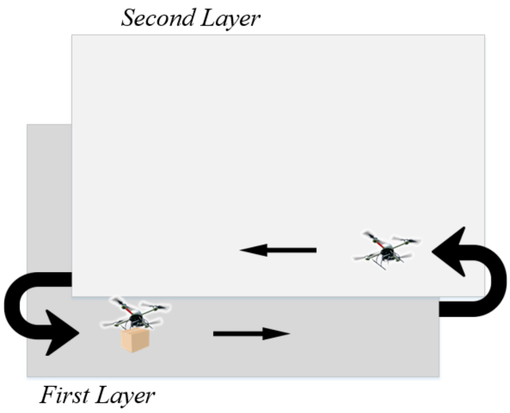
- Highway, in which robots without parcels are separated from robots with parcels on the same plane (Figure 33b);
- Two-layered, in which robots with parcels are separated from robots without parcels on two planes (Figure 33c).



(a)



(b)



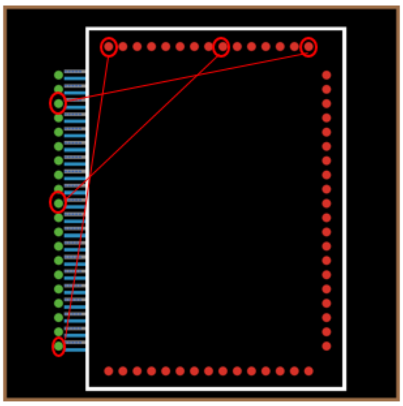
(c)

Figures 33a-33b-33c: Traffic design configurations: (a) Mixed Traffic, (b) Highway, (c) Two-layered

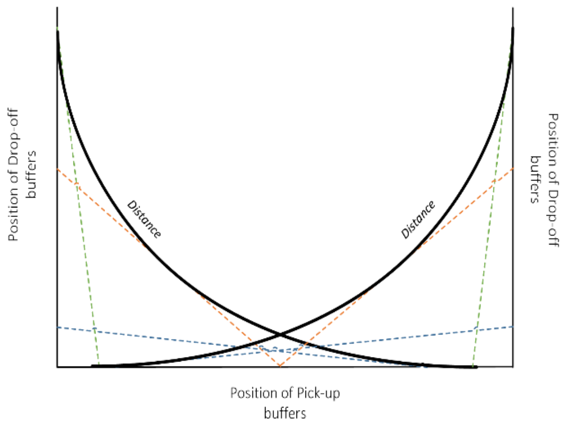
Mixed Traffic. In *Mixed Traffic*, robots with parcels and without parcels share the same field (see Figure 33a). In this configuration, the number of interferences between loaded and unloaded robots increases, with higher number of brake points and dodging of robots. Evidently, the number of interferences depends on the path planning and resource allocation algorithms and the layout of the sorting hub. However, independently on the layout or path planning and resource allocation algorithms employed, a comparable higher amount of interferences and deviations remains in this traffic situation. Higher number of interferences may entail higher service times to deliver a parcel. Nevertheless, in this traffic configuration, robots are likely to travel short distances and, consequently, have low idle periods.

In Chapter 4, we have described the min-max heuristics, based on waiting time of parcels, for the allocation of robots to pick-up buffers. Using the min-max heuristics algorithm, robots choose to either minimize the maximum waiting time or to follow randomly one pick-up buffers within a certain distance range. The distance robots travel depends on the traffic configuration. In Mixed Traffic, robots select the pick-up buffer to follow once they have dropped off their parcels. After they release their parcels, robots follow a pick-up buffer located at a distance, from their current drop-off buffer, lower than a maximum predefined distance. In Figure 34a, the maximum distance corresponds to the red lines connecting drop-off buffers to pick-up buffers. It is important to stress that robots do not follow precisely the pick-up buffer with maximum distance from their position, but one of the pick-up buffers located at a distance shorter than the maximum predefined distance (red line). By doing so, robots follow a curved trajectory, as can be observed in Figure 34b.

This is also valid for robots at drop-off buffers on the right side. These robots will select one of the pick-up buffers within a maximum distance and defined angle from their position. In this way, we are able to partially strive to optimize the distribution of robots over pick-up buffers and partially minimize the maximum waiting time of parcels, which is likely to match with the waiting time of heavy and high volume parcels that require the joint effort of multiple robots.



(a)



(b)

Figures 34a-34b: Method to calculate max distance for min-max heuristics algorithm in Mixed Traffic

Highway. In *Highway*, robots with parcels and without parcels are separated on the same plane (see Figure 33b). Loaded robots travel from pick-up buffers to destination of the parcels, after which point they take a path outside the transport field.

The only remaining interferences between robots with parcels and robots without parcels occur when unloaded robots enter the queue of the pick-up buffers, while loaded robots leave queues with their parcels. Therefore, the number of interferences between loaded and unloaded robots is minimal. It is important to underline that robots leaving the transport area do not follow a fixed path, but follow reference points. This entails that robots do not move in one-dimension, but they are also able to overtake other robots, like in case of failed robots in the fixed path. Reference points are located at the corners outside the transport field (in Figure 33b represented by the turning arrows).

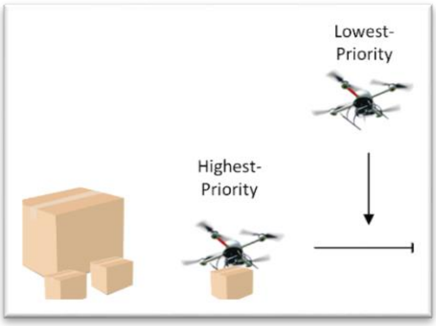


Figure 35: Priority scheme for collision avoidance in Highway

We have decided to design the *Highway* in this way, with a path outside the transport area, in order to increase the safety of robots by drastically reducing the interferences between them. Therefore, this traffic design compensates a loss in distance travelled by robots, which is likely to be longer than in other configurations, with an increase of safety. Furthermore, unloaded robots in this configuration do not face side collisions or frontal collision on different directions, which are time-consuming to avoid, given they travel behind each other. For this reason, the service time of parcels might be reduced. Moreover, this configuration increases the predictability of the system, with the behaviors of robots becoming easier to understand and control.
As already stated, the only remaining interferences between loaded and unloaded robots occur when unloaded robots travel towards the pick-up buffers through the highway, while loaded robots are leaving the exit queue. These interferences constitute potential collisions between robots that are handled by assigning robots with different priorities. Robots with parcels are assigned with the highest priority, while robots without parcels travelling through the highway are assigned with the lowest priority (see Figure 35). Therefore, high-priority robots do not brake to dodge their

obstacles, predicting a brake of the other robots. Oppositely, low-priority robots brake to let loaded robots move towards the transport field, then accelerate to reobtain speed and move forward towards pick-up buffers.

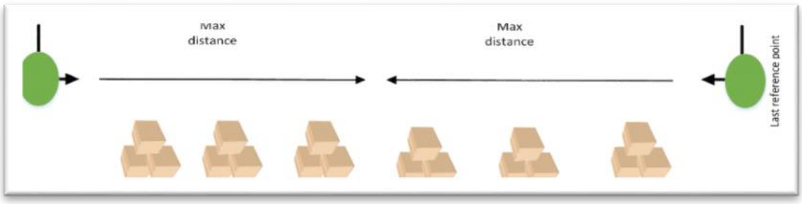


Figure 36: Method to calculate max distance for min-max heuristics algorithm in *Highway*

Regarding the allocation of robots to the pick-up buffers, the maximum distance a robot can travel to reach a station is measured differently than in the *Mixed Traffic* configuration. Here, in fact, robots choose a pick-up station only once they arrive at the last reference points, being the corner points with the shortest distance from pick-up buffers. As can be observed in Figure 36, the maximum distance robots can travel from the last reference points (green circles) is calculated in such a way that robots do not have to dodge each other before reaching the pick-up buffers. Therefore, robots can choose a pick-up buffer that is located at a distance shorter than the indicated maximum distance.

Two-layered. In *Two-layered*, robots with parcels and without parcels are separated on two different planes (see Figure 33c). In this traffic configuration, robots with parcels travel on the first plane, whereas robots without parcels travel on the second plane. In short, once obtained a parcel, robots move towards the appropriate drop-off buffers to release their parcels. After that point, if we consider robots as UAVs, they fly upwards to the second plane and backwards towards a pick-up buffer. When the pick-up buffer is reached, they fly downwards to collect a parcel. Instead, if we consider robots as UGVs, once released a parcel, each robot takes a ramp that connects the first floor with a second floor. They travel on the second floor without a parcel to reach the other ramp that connects the second floor with the queues of pick-up buffers. Therefore, once reached the appropriate ramp, robots move downwards to enter the queue of a pick-up buffer and, subsequently, collect a parcel.

The *Two-layered* traffic configuration does not differ from the *Mixed Traffic* configuration with regard to the way robots are allocated to pick-up buffers. However, using this configuration, the number of interferences is reduced by separating the different entities (loaded-unloaded robots). Therefore, the distance travelled by robots is almost the same, but the service time is likely to be reduced. In comparison to the *Highway*, there are more frontal and side collisions to avoid, but the distance travelled by robots is presumably shorter.

In the next paragraph, we describe the experimental designs that we use to compare different design alternatives.

5.2 Experimental Designs

As earlier stated, *experimentation* is the phase where inferences can be obtained by conducting simulations with changing input variables and observing their effects on performance indicators. In Chapter 4, the input variables for this model were listed. The alteration of these parameters can provoke larger or smaller effects on performance indicators. However, altering all these parameters would imply an endless number of combinations/scenarios and would not be useful to address our research questions. Therefore, we have to decide what parameters to keep constant and what parameters to change, according to the aim of this research.
It is important to underline again that this experimental phase focuses on:

- ***Impact of cooperative transport on system effectiveness***: the objective is to assess the impact of cooperative transport on the performance indicators that concern the amount of parcels transported per unit of time by robots, such as throughput, robot performance, service time, waiting time, distance travelled. Therefore, we aim at determining the impact of increasing amount of heavy and high volume parcels on these performance indicators and inferring the tolerable variation of these parcels.
- ***Impact of cooperative transport on congestion:*** the objective is to assess the impact of cooperative transport on workspace, by looking at another performance indicator, being congestion. Congestion is calculated as the ratio between the average speed of robots and the maximum speed of robots. Therefore, this performance indicator can give us insights into the number of interferences brought about by cooperating robots, which travel at reduced speed. In this way, we are able to determine whether cooperative robots require larger workspaces to act or not in comparison to robots transporting light and low volume parcels.
- ***Impact of cooperative transport on fault tolerance:*** the objective is to assess the impact of cooperative transport on fault tolerance, which was

defined in Chapter 2 as the ability of a system to keep operating even in the presence of failure of one (or more) of its elements. Therefore, we want to design disruptive scenarios, where a number of robots and pick-up buffers fail and evaluate the impact on the system. In particular, we are interested in assessing the impact of robots moving heavy and large loads on fault tolerance. In fact, when a robot fails during the transportation of these loads, the whole formation collapses, thus having a larger impact on fault tolerance in comparison to robots transporting light and small loads.

These objectives guide us towards the design choices to make in order to create meaningful scenarios and address the mentioned problems. Accordingly, we have decided on the input parameters to keep constant and the parameters to vary in order to address the investigated problems profitably and reduce the number of potential combinations available to the minimum required for the scope of this research. The input parameters that we have determined to keep constant are displayed in Table 12.

Table 12: Fixed input parameters and values

Input Parameters	Values
Number of pick-up buffers	20
Number of drop-off buffers	50
Maximum capacity of pick-up buffers	1 parcel
Speed of robots	2 m/s
Speed of cooperative robots	1 m/s
Position of drop-off and pick-up buffers (sorting hub layout)	Fixed positions[2]
Temporal interval to load a parcel	4 s
Temporal interval to unload a parcel	5 s
Arrival parcel distribution	Uniform

The number of pick-up and drop-off buffers is unchanged, due to the scalability and flexibility of the developed system. In fact, when the number of buffers is increased or reduced, more or less robots can be used, with none or little differences in the average results. The maximum capacity of pick-up buffers is one parcel, because we assume that operators at these stations can only load one parcel on robots at the time. Prime Vision has provided us with the data regarding the speed of robots with small loads or unloaded. We assume that when robots travel in a formation, the speed is reduced to

[2] See paragraph 4.5 of Chapter 4 for the layout of the sorting hub

give stability to the formation and to the parcels placed on them. One meter per second seems a reasonable speed to achieve these objectives. As argued in Chapter 2, we are not focusing on the optimization of sorting hub layout, but on the optimization of the operations of sorting systems. Therefore, the position of drop-off and pick-up buffer is voluntarily kept constant. PostNL has provided the information with regard to the temporal interval to load and unload parcels, thus these parameters are also constant.

Consequently, the input parameters that we have decided to vary in the experimental designs are namely *number of robots*¸ *percentage light-low volume parcels* (hence, *percentage heavy-high volume parcels*), *traffic design alternatives*, *number of faulty robots, number of faulty pick-up buffers* and *with or without assistance mechanism.*

In consideration of the workspace, the number of robots can range from 100 to maximum 200. Mathematically, this can be calculated considering robots as circles with a certain radius (in this case, we assume the radius to be equal to 0.95 m). Accordingly, the workspace required by 200 robots is 709.49 m^2, excluding the space required by pick-up and drop-off buffers (see Appendix L). Therefore, the sorting system can tolerate this number of robots. The percentage of light and low-volume parcels can range from 100% to 85%. This entails that the percentage of heavy and high-volume parcels can range from 0% to 15%. At present, the percentage of heavy and high-volume parcels is between 5 and 10%, according to PostNL suggestions. In the future, given the rise of e-commerce an increase of this percentage is expected. Therefore, this range is appropriate to assess the impact of cooperative transport on current state and future state of postal companies and compare it with a situation with only light and low-volume parcels.

As described earlier, three traffic design alternatives can be employed to control the traffic inside the transport area, which are *Mixed Traffic*, *Highway* and *Two-Layered*. We can alter these traffic design alternatives to compare them based on the system effectiveness and level of congestion provided.

By altering the input parameters described above, we are able to build six experimental designs (Table 13). Certain experiments contain 120 combinations of values. The first three scenarios are used to address the first two objectives of the research, being observing the impact of cooperative transport on system effectiveness and congestion. With regard to the last objective of the research, i.e. addressing the impact of cooperative transport on fault tolerance, we alter the number of faulty robots and pick-up buffers, with or without assistance mechanism. Moreover, we have established that Mixed Traffic is the most adequate traffic configuration to use in order to build meaningful disruptive experiments. In Mixed Traffic, the level of congestion is higher than in the other two traffic designs, given that loaded and unloaded robots share the same field (see results in paragraph 5.3.2). Therefore, the failure of robots and pick-up buffers might have a larger impact in this configuration than in others. The number of robots is fixed to 150, given that Mixed Traffic poorly tolerates higher number of robots, as will be shown later in the results. The percentage of heavy and high volume parcels is kept constant to 10%, which corresponds to the current maximum percentage of these loads in PostNL. The number of faulty robots can range from 1 to 5 (from 0.67 to 3.3% failures). The causes of failures are manifold (e.g. software, hardware, energy failures); therefore, although improbable, this number of failures might be observed in practice. The number of faulty pick-up buffers ranges from 1 to 4 (from 5 to 20%). Finally, in the last scenario, we include the assistance mechanism, with robots placed outside the area coming into the field to support the faulty robots.

Table 13. Experimental Designs

Nº	Number of robots	% light-low volume parcels	% heavy-high volume parcels	Traffic design	# faulty robots	# faulty pick-ups	With(out) assistance
1	100 – 150 – 200	100 – 95 – 90 – 85	0 – 5 – 10 – 15	Mixed Traffic	-	-	-
2	100 – 150 – 200	100 – 95 – 90 – 85	0 – 5 – 10 – 15	Highway	-	-	-
3	100 – 150 – 200	100 – 95 – 90 – 85	0 – 5 – 10 – 15	Two-Layered	-	-	-
4	150	90	10	Mixed Traffic	1– 2 – 3 – 4 - 5	-	No
5	150	90	10	Mixed Traffic	1 – 2 – 3 – 4 -5	1-2-3-4	No
6	150	90	10	Mixed Traffic	1 – 2 – 3 – 4 – 5	-	Yes

It is important to indicate the time of the simulation runs, which we have decided to be half hour. Assuming that four ticks in NetLogo correspond to one second, we will run the simulation for 7200 ticks. However, *in order to obtain steady state results, thus eliminating transient state results, we will use a longer run-time (8200 ticks) and delete the results from the first 1000 ticks.* By means of observation, we have decided to eliminate the data collected in the first 1000 ticks for all performace indicators. In some experiments, we have noticed that transient state lasts less than in others. For instance, in the experiments with 200 robots and in Highway, steady results are achieved later. Among the several performance indicators, service time and waiting time are more difficult to measure. This can be seen in Figures 39a-39b-39c, where steady results in some scenarios are obtained after approximately 1500 ticks. In the next paragraph, we display and interpret the results from the experimental designs. Although it is hard to determine a reasonable run length and we are wasting resources/time, the elimination of the initial data provides almost in all experiments steady results.

5.3 Results from experiments

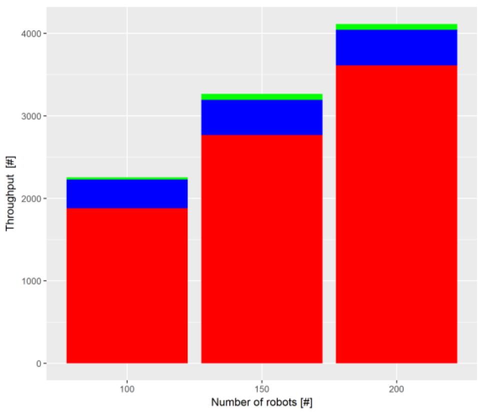
In this paragraph, we report and analyze the results obtained from various experimental designs, shown in Table 13. The first three experimental designs are used to investigate the impact of cooperative transport on system effectiveness and on congestion; while the other three experimental designs are used to investigate the impact of cooperative transport on system fault tolerance. In the first three experimental designs, we only present the results obtained with 100% light and low volume parcels and compare these with the results obtained in the scenarios with 90% light and low volume parcels. Therefore, we omit results obtained in the scenarios with 95% and 85% light and low volume parcels. The reason behind this choice is that the impact of cooperative transport becomes highly noticeable with 90% compared to 95%. This impact becomes even more evident with 85% light and low volume parcels, but the conclusions inferred would be the same. However, these results are reported in the Appendices M, N, O, P for consultation.

Furthermore, it is important to remark that these results are obtained after running the various simulations for 8200 ticks. However, we have decided to eliminate data from the first 1000 ticks, in order to obtain steady-state results. Therefore, results are reported over a timeframe of 7200 ticks. This timeframe, assuming the value of 4 ticks per seconds, corresponds to 30 minutes in real-life. Moreover, for the majority of the results, we have run the same simulations, with equal combinations of values, 10 times. For other results, like for the fourth experimental design, we have run the same simulations 20 times in order to have insightful results. The results from the experiments are displayed using different types of figures, such as bar charts, box plots and scatter plots, and tables. When the standard deviation is not visible, bar charts are preferred to box plots.

5.3.1 Impact of cooperative transport on system effectiveness

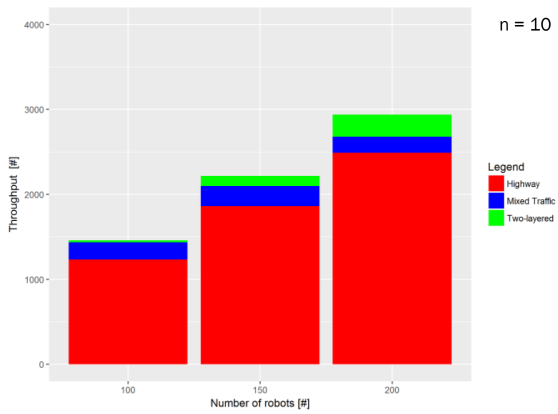
In Chapter 2, we have linked the sorting operations to a maximal flow transportation problem, where the objective is to maximize the parcel volume onto outbound trucks. Correspondingly, one of the most important performance indicators for parcel operators is *throughput*, i.e. total amount of sorted parcels.

This indicator is measured in the first three experimental designs with changing number of robots, percentage of light/low volume and heavy/high volume parcels and different traffic management approaches. Figure 37a displays a bar chart where the throughput is measured in a scenario with 100% light and low volume parcels. In this figure, the x-axis corresponds to the number of robots and the y-axis to the throughput. The different colors are used to distinguish and compare the different traffic design alternatives. Noticeably, the throughput grows with increasing number of robots in all three traffic configurations.



	Mixed	*Highway*	*2Layered*
Mean 100	2226.7	1879.4	2255.1
Mean 150	3193.6	2768.6	3267.3
Mean 200	4043.2	3610.4	4113.5
St.dev 100	9.59	7.92	10.09
St.dev 150	7.05	15.07	7.41
St.dev 200	12.02	14.52	18.5

(a)



	Mixed	*Highway*	*2Layered*
Mean 100	1436.4	1234.4	1458.8
Mean 150	2098.2	1861.8	2218.5
Mean 200	2678.7	2491.1	2938
St.dev 100	51.35	31.3	37.9
St.dev 150	33.02	33.47	52.2
St.dev 200	59.69	36.64	45.7

(b)

Figures 37a-37b: Throughput with 100% (a) and (b) with 90% small parcels

Furthermore, the throughput is higher in Mixed Traffic and Two-Layered in comparison to the Highway in any experiment. The difference in terms of throughput between Mixed Traffic and Highway is 18.5% with 100 robots, 15.3% with 150 robots and almost 12% with 200 robots. This demonstrates that the throughput in the Highway increases more with increasing number of robots than in Mixed Traffic. However, Mixed Traffic guarantees higher throughput in comparison to Highway. In this scenario, Mixed Traffic and Two-Layered do not show a significant difference in throughput.

Figure 37b shows the throughput obtained in a scenario with 90% light and low volume parcels; hence, in this scenario, there is 10% heavy and high volume parcels. When observing the bar chart of figure (b), it is apparent how cooperative transport affects adversely the amount of parcels delivered in the considered timeframe. This result was predictable, given that, in this scenario, four robots are used to transport 10% of the parcels. Therefore, the throughput decreases heavily as we are using more resources to transport heavy and high volume parcels. Furthermore, cooperative transport has the greatest negative impact on Mixed Traffic, especially with increasing number of robots. In fact, the mean of throughput decreases by 55% with 100 robots, 52% with 150 robots and 50% with 200 robots in Mixed Traffic. In comparison, in Two-Layered, the mean of throughput decreases by 54% with 100 robots, 47% with 150 robots and 40% with 200 robots.

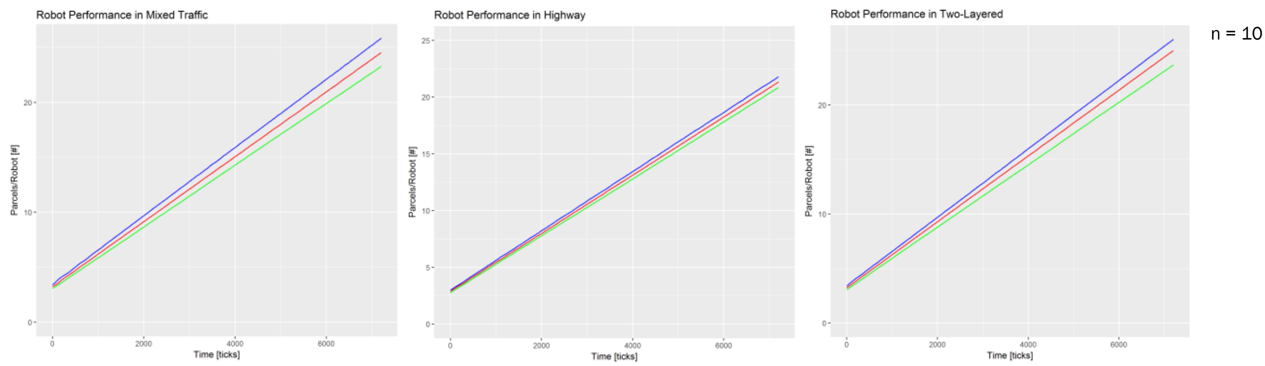
As can be observed, in comparison to the previous scenario with 100% light and low volume parcels, the difference in throughput reduces as the number of robots increases. One conclusion from this result is that cooperative transport requires a higher number of robots to achieve the same results obtained in the first scenario, with 100% light and low volume parcels. This is caused by the fact that robots wait long temporal intervals at pick-up stations to build the formations of robots. The higher the number of robots, the higher the probability formations of robots are built. Comparing the results from the two scenarios, we can observe that in Mixed Traffic we need around 60 robots more in the second experiments to achieve the same throughput obtained with 100 robots in the first scenario. While, we need almost 90 robots more in the second scenario to obtain the throughput achieved by 150 robots in the first scenario in Mixed Traffic. In Highway, we need around 50 robots more to achieve the same throughput obtained in the first scenario with 100 robots. While, we need around 70 robots more to achieve the same throughput in the first scenario with 150 robots. The results obtained in the Two-Layered follow a similar trend with the results obtained in the Highway. Therefore, we can conclude that Mixed Traffic requires a higher number of robots in comparison to Highway and Two-Layered to achieve the same throughput of the first scenario.

With regard to the fulfilment of the minimum throughput (5000-8000) required for this system under the input parameters analyzed, this system is able to satisfy this constraint granted that an elevated number of robots is used. In a scenario with 100% light and low volume parcels, using 100 robots in a Two-layered configuration (best case scenario), we are only able to achieve 2255 parcels in half hour (approx. 4510 parcels/hour). Therefore, the constraint is not satisfied using only 100 robots. In the same scenario but with 150 robots, the system is able to reach a throughput of 3267 parcels / half hour (on average). This means that the system is able to meet the constraint, achieving around 6500 parcels / hour. Using 200 robots in Two-layered, the system is able to exceed the throughput required, completing around 8200 sorting tasks in one hour. In Mixed Traffic, similar results are obtained in a scenario with only small parcels. In Highway, instead, the maximum throughput achieved is of roughly 7200 parcels / hour. This means that in this traffic configuration, the system is able to exceed the minimum threshold of 5000 parcels / hour, but it is not able to reach the maximum threshold of 8000 parcels / hour using 200 robots. In a scenario with 90% light and low volume parcels, in Two-layered (best case scenario), the system is able to exceed 5000 parcels / hour using 200 robots. Additional robots are required to achieve a throughput of 8000 parcels / hour. Overall, the system shows that it has the potential to meet the throughput capacity required by postal operators. These results can be further improved by reducing the distances between pick-up and drop-off buffers (layout optimization), increasing the input buffers (reducing waiting time of parcels) or implementing sorting systems on different floors.

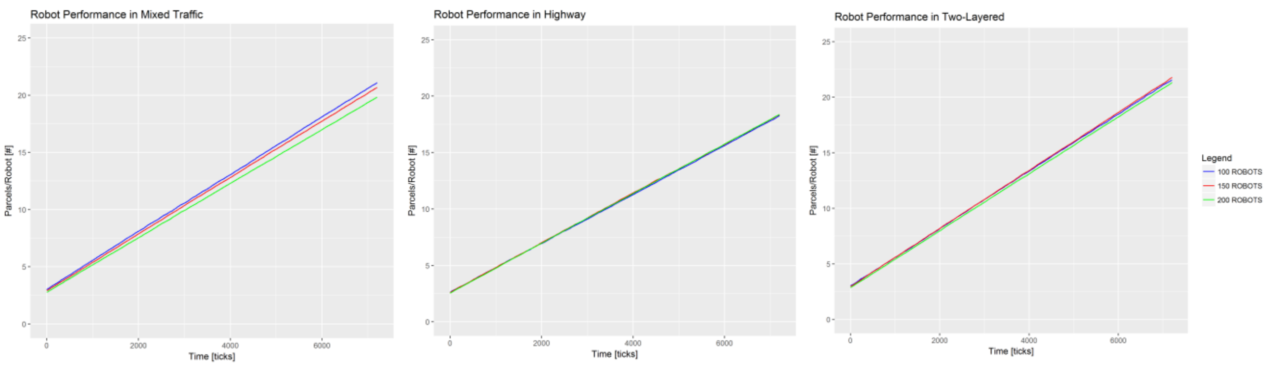
Robot performance is another extremely useful indicator for parcel operators, considering that the amount of parcels to sort fluctates over time. Therefore, using this indicator, parcel operators can decide to add a precise number of robots to the system in order to satisfy the changing demand.

When we inspect the performance of robots, meaning the number of parcels transported by robots (Figures 38a-b-c-d-e-f), in Mixed Traffic and with 100% light and low volume parcels (line chart 38a), we can observe that robot performance decreases with increasing number of robots. The three lines diverge progressively over time, thus increasing the gaps between them. This entails that after a certain time period, using more robots will produce lower throughput. In the scenario with Highway and 100% light and low volume parcels (line chart 38b), the difference between the three lines is limited, thus highlighting how this configuration can tolerate higher number of robots in comparison to Mixed Traffic. In Two-Layered and with 100% light and low volume parcels (line chart 38c), the line reclines more with 200 robots, while the difference between the lines obtained with 100 robots and 150 robots is more restricted. This results in a higher ability to cope with 100 and 150 robots than with 200 robots. As we will see later, the differences in robot performance between these three traffic configurations is likely to be caused by the different levels of congestion obtained, which lead to higher service times.

Interestingly, when inspecting the trend with 90% light and low volume parcels, we can immediately notice the alteration of robot performance. As we have inferred earlier, scenarios with cooperative transport demand for higher number of robots. Therefore, what we see in the line charts 38d, 38e and 38f is a decline of the line with 100 robots and an increase of the lines with more robots. In the scenario with Mixed Traffic and with 90% light and low volume parcels, the difference between the lines with 100 and 150 robots has shortened, while there is still a visible gap with 200 robots. One conclusion from this result is that Mixed Traffic tolerates efficiently up to 150 robots in this scenario.



(a) 100% Mixed Traffic (b) 100% Highway (c) 100% Two-Layered



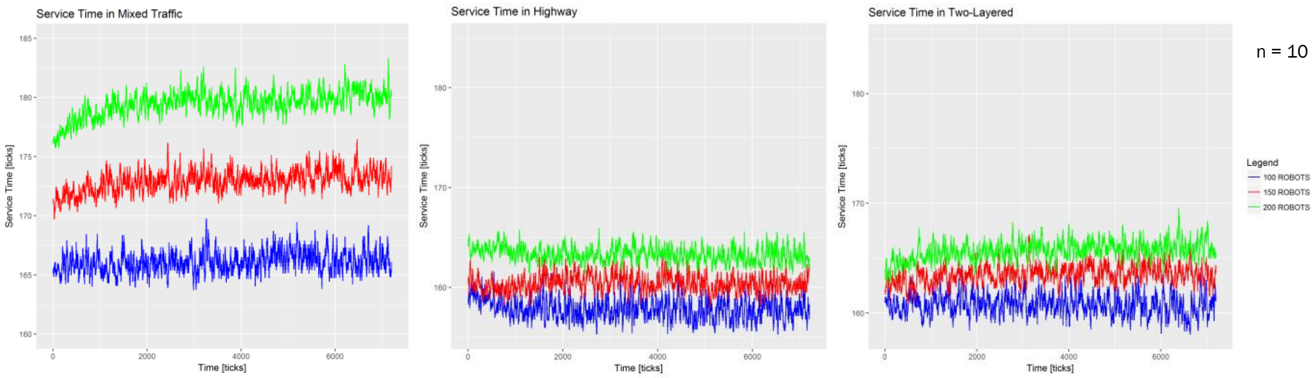
(d) 90% Mixed Traffic (e) 90% Highway (f) 90% Two-Layered

Figures 38a-b-c-d-e-f: Robot performances in the three traffic designs with 100% and 90% small parcels

It is equally striking to notice the trend in Highway and Two-Layered with increasing percentages of heavy and high volume parcels. In these configurations, the performance of robots does not differ when more robots are introduced in the system. This means that adding extra robots would lead to invariable performance of robots and regularly higher throughput. In a scenario with 85% heavy and high volume parcels (see Appendix M for results), the trend is confirmed in Mixed Traffic where the gap between the lines corresponding to 100 and 150 robots do not differ significantly, while a wider difference is visible with 200 robots. This further confirms the conclusion that

Mixed Traffic poorly tolerates over 150 robots with high percentages of heavy and high volume parcels. In the same scenario with 85% big loads in Highway and Two-layered, the robot performance with 100 robot decreases below the level of the lines with 150 and 200 robots, suggesting that these configurations favor increasing number of robots.

Two main drivers, being *Service Time* and *Percentage Distance Travelled Idle*, influence significantly the results observed above, with regard to throughput and robot performance. Service time refers to the time robots take to transport parcels from pick-up buffers to drop-off buffers. As can be observed looking at Figures 39, the service time in Mixed Traffic (Figure 39a) is higher than the service time in Highway (Figure 39b) and Two-Layered (Figure 39c). This result is due to the absence of separation of loaded and unloaded robots in the Mixed Traffic, while in the other two traffic approaches robots without parcels do not interfere with robots transporting parcels.

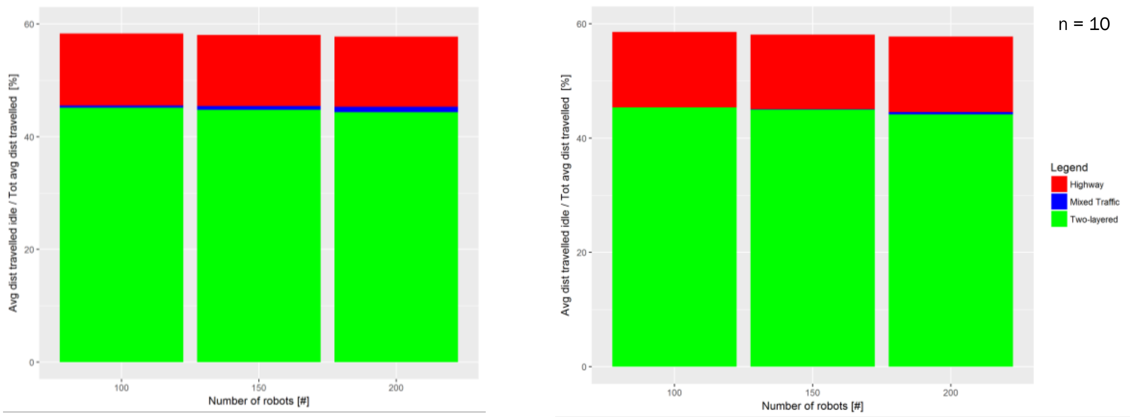


(a) (b) (c)

Figures 39a-39b-39c: Service time in Mixed Traffic (a), Highway (b) and Two-Layered (c) with 100% small parcels

Furthermore, in all three configurations, the service time increases with increasing number of robots, as a result of the increase of congestion in the traffic area. This increase is more evident in Mixed Traffic because of the lack of separation of the different entities. The service time slightly changes between Highway and Two-Layered, as again in both cases robots with parcels do not find obstacles along their motion. Figures 39a-b-c display the results related to service time in the scenarios with 100% light and low volume parcels. In this scenario all robots travel at 2 m/s. The time is reported in ticks, which we assume have a value of a quarter of seconds. Therefore, the mean service time in Mixed Traffic is equal to 41.125 seconds with 100 robots, 43 seconds with 150 robots and 45 seconds with 200 robots. In Highway, the mean service time is 39.375 seconds with 100 robots, 40 seconds with 150 robots and around 41 seconds with 200 robots. In comparison to the service time in Highway, in Two-Layered the service time is slightly higher, due to the small difference in level of congestion.

Cooperative transport influences the service time, considering that robots in formation travel at 1 m/s. Considering scenarios with 90% heavy and high volume parcels, the mean service time to transport big parcels doubles the time required to transport small parcels, as for the speed of robots. Therefore, it will take around 93 seconds on average to transport heavy and high volume parcels in Mixed Traffic, around 82 seconds on average in Highway and around 85 seconds in Two-Layered (results can be found in the Appendix O). Cooperative transport marginally influences the service time for light and low volume as robots with small parcels need to reduce their speed when travelling behind formations of robots and to avoid collisions with them. However, this increase is not considerable (see results in Appendices M, N, O, P).

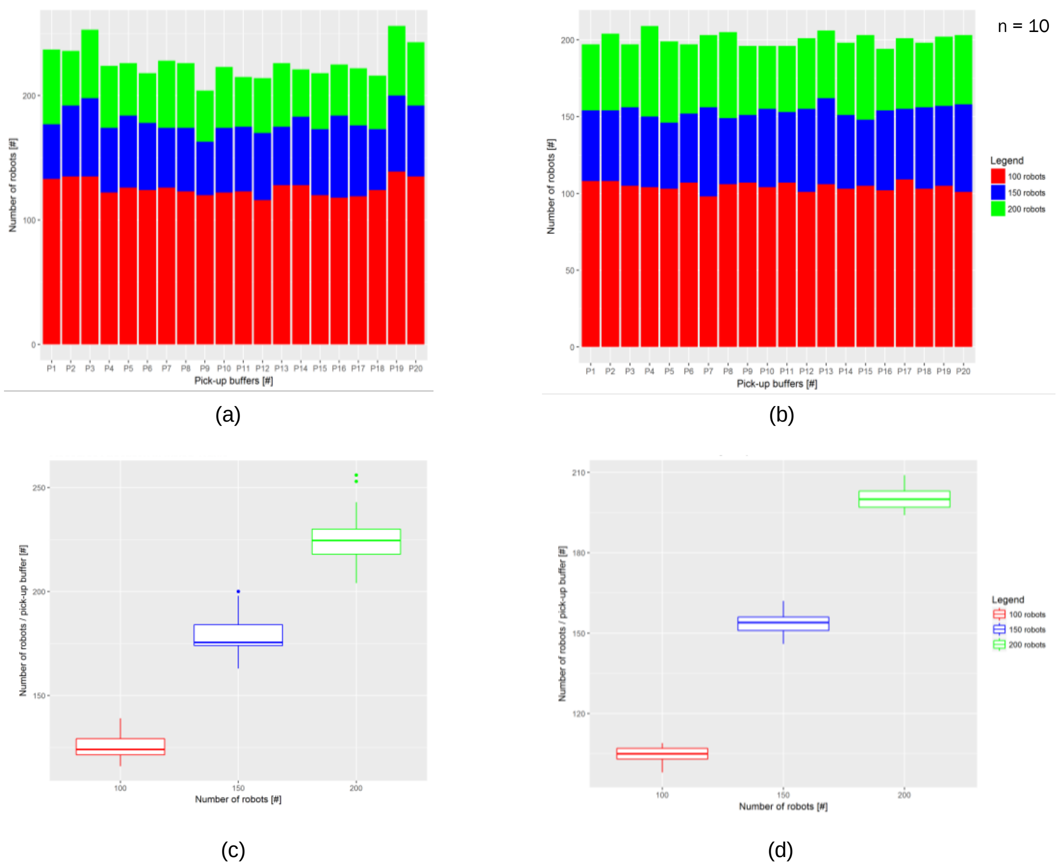


(a) (b)

Figures 40a-40b: Distance travelled in the three traffic designs with 100% (a) and 90% (b) small parcels

Percentage distance travelled idle is another driver for throughput and indicates the percentage distance travelled by robots without parcels (i.e. idle). This is measured by calculating the distance travelled by robots without parcels and dividing this amount by the total robots distance travelled. Figures 40a-40b display the percentage distance travelled idle by robots in a scenario with 100% and 90% light and low volume parcels. As we can see, there is no significant difference between distances travelled in the different scenarios.

This was another expected result, considering that the distance should not change with or without formations of robots. Furthermore, the results obtained in Mixed Traffic and Two-Layered are positive, considering that robots travel more with parcels than without parcels. Also predictably, there are no differences between these two traffic configurations in terms of distance travelled by robots. In Highway, robots travel almost 10% more without parcels than with parcels. This result was also foreseeable, considering that in Highway robots take a path outside the transport field and follow it until destination. Therefore, the travelled route to return to pick-up buffers is longer than the travelled route to transport parcels. These results also validate the heuristic algorithm employed for the resource allocation.



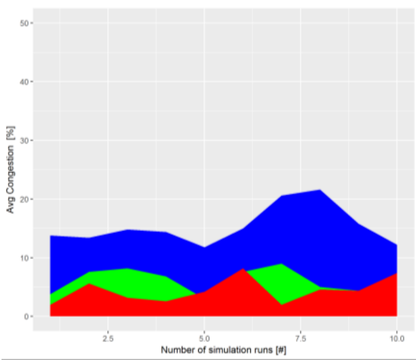
Figures 41a-b-c-d: Histograms and boxplots for resource allocation in Mixed Traffic (a) and in Highway (b) with 100% small parcels

In fact, as can be observed in histogram in Figures 41a and 41b, robots distribute finely across the twenty pick-up buffers (x-axis). These histograms are obtained after calculating the mean of the results achieved after multiple runs, therefore the results linearize after running many simulations. For this reason, we also show the boxplots 41c and 41d, which demonstrate that the standard deviations are not overly large, proving the adequacy of the heuristic algorithm.

It must be noticed that we have only displayed the results for Mixed Traffic and Highway. Indeed, it is not important to show the results for Two-Layered, as these are the same of the ones obtained in Mixed Traffic. The algorithm only differs in the calculation of the maximum distances between Mixed Traffic (or Two-Layered) and Highway.

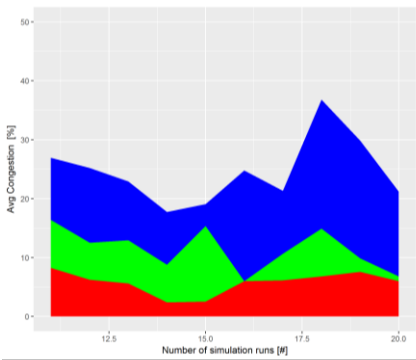
5.3.2 Impact of cooperative transport on congestion

Congestion is another important performance indicator for a multi-robot parcel sorting system because it gives insights into the number of interferences between robots and into the workspace required by the system to work efficiently. As earlier stated, congestion is measured as the ratio between average speed and maximum speed of robots. As for system effectiveness, results pertaining to congestion are collected in the first three experimental designs. Looking at the results analyzed in the previous paragraph and especially at service time, we can already anticipated higher levels of congestion in Mixed Traffic in comparison to the other two traffic design alternatives.



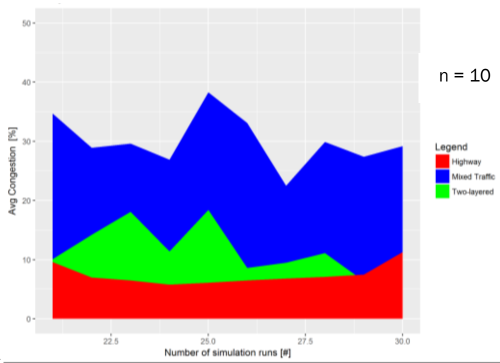
	Mixed	Highway	2Layered
Mean 100	16%	4%	6%
St.dev 100	0.031	0.02	0.02

(a)



	Mixed	Highway	2Layered
Mean 150	25%	6%	12%
St.dev 150	0.053	0.018	0.032

(b)



	Mixed	Highway	2Layered
Mean 200	30%	7%	12%
St.dev 200	0.042	0.016	0.039

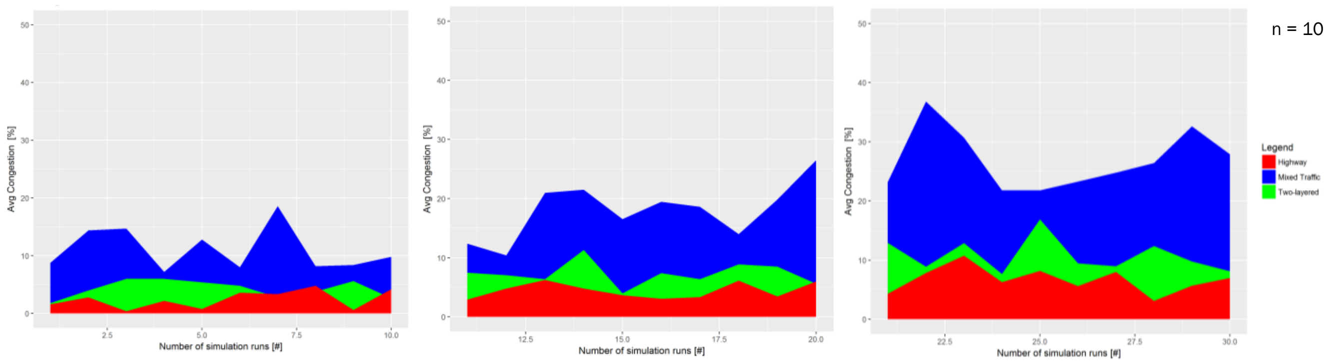
(c)

Figures 42a-42b-42c: Congestion with 100 (a), 150 (b) and 200 (c) robots and with 100% small parcels in the three traffic designs

Figures 42a-42b-42c show the different degree of congestion in the three different traffic scenarios with increasing number of robots and with 100% light and low level parcels. In these graphs, the x-axes represent the number of simulation runs performed, while the y-axes represent the average percentage of congestion. Seemingly, in all three scenarios, the level of congestion in Mixed Traffic is the highest, while the level of congestion in Highway is the lowest. In comparison to the level of congestion in Highway, in Mixed Traffic the level of congestion is 12% higher with 100 robots, 19% higher with 150 robots and 23% higher with 200 robots. The level of congestion in Mixed Traffic almost doubles when the number of robots increases from 100 to 200. Consequently, robots decelerates many times to avoid other robots in this traffic configuration. In Highway, the congestion only rises by 3% from 100 robots to 200 robots. This entails that this traffic configurations tolerates well high number of robots. In Two-Layered, the level of congestion increases from 6% to 12%. Interestingly, in this traffic configuration, the level of congestion stabilizes after 150 robots. The level of congestion is higher in Two-Layered than in Highway, because in the latter the behavior of robots when travelling on the path outside the field becomes predictable as less collisions are faced. In Two-layered, although loaded and unloaded robots are separated, robots are still facing side and frontal collisions on both planes.

It is important to underline that higher congestion (hence higher service and return times) does not necessarily entail lower throughput, particularly with 100% light and low volume parcels. In fact, earlier we have seen that, although the high degree of congestion, Mixed Traffic provides good throughput. By reason of the number of pick-up buffers in the system (20), we can have up to 5 robots per buffer with 100 robots, 7.5 robots per buffer with 150 robots and 10 robots per buffer with 200 robots. Considering also the relatively low percentage of distance travelled idle and the time interval robots spend at pick-up buffers before being loaded with a parcel, generally robots need to wait in a queue before having the possibility to access a buffer. Therefore, even if the congestion is higher in Mixed Traffic in comparison to the other traffic configurations, this does not have an evident impact on throughput, when there are only light and low volume parcels. This conclusion is confirmed by looking at the results of waiting time (see Appendices M, N, O, P for graphs on waiting time). Indeed, the waiting time of parcels at the pick-up buffers does not change significantly between Mixed Traffic and Two-Layered, although the great difference in congestion. This means that even if the service and return time are higher in Mixed Traffic, due to the higher congestion, there is always an adequate number of robots at pick-up buffers.
Oppositely, with increasing percentages of heavy parcels, the pick-up buffers require more robots. In these conditions, congestion will have higher impact on throughput, as can be observed in Figure 37b. This result is proven by the increased difference in waiting time between Mixed Traffic and Two-Layered (see Appendix O).

In comparison to the results obtained in the scenarios with 100% light and low volume parcels, when the percentage of heavy and high volume parcels increases, the level of congestion decreases in every scenario (see Figures 43a-43b-43c). The number of interferences between robots is indeed lower, as a result of the absence of interferences between robots in formations.



	Mixed	Highway	2Layered
Mean 100	11%	2%	4%
St.dev 100	0.036	0.015	0.014

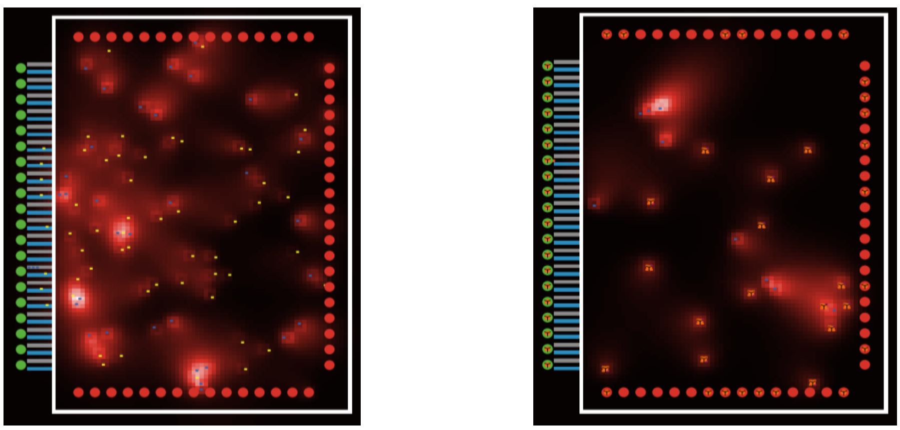
	Mixed	Highway	2Layered
Mean 150	18%	4%	7%
St.dev 150	0.045	0.013	0.019

	Mixed	Highway	2Layered
Mean 200	27%	7%	11%
St.dev 200	0.048	0.02	0.027

(a) (b) (c)

Figures 43a-43b-43c: Congestion with 100 (a), 150 (b) and 200 (c) robots and with 90% small parcels in the three traffic designs

Reasonably, in the scenarios with zero big loads, all robots are separated / individual entities moving into the space; whereas, in the scenarios with for instance 10% big loads, aggregates of four robots move into the space as single large entities, reducing the coverage of the space. Consequently, the higher the percentage of heavy parcels, the smaller the space occupied by robots and, therefore, the lower the congestion in the transport field. This effect of cooperative transport on congestion can be observed by implementing a heat-map featuring the concentration and coverage of robots into the space.



(a) (b)

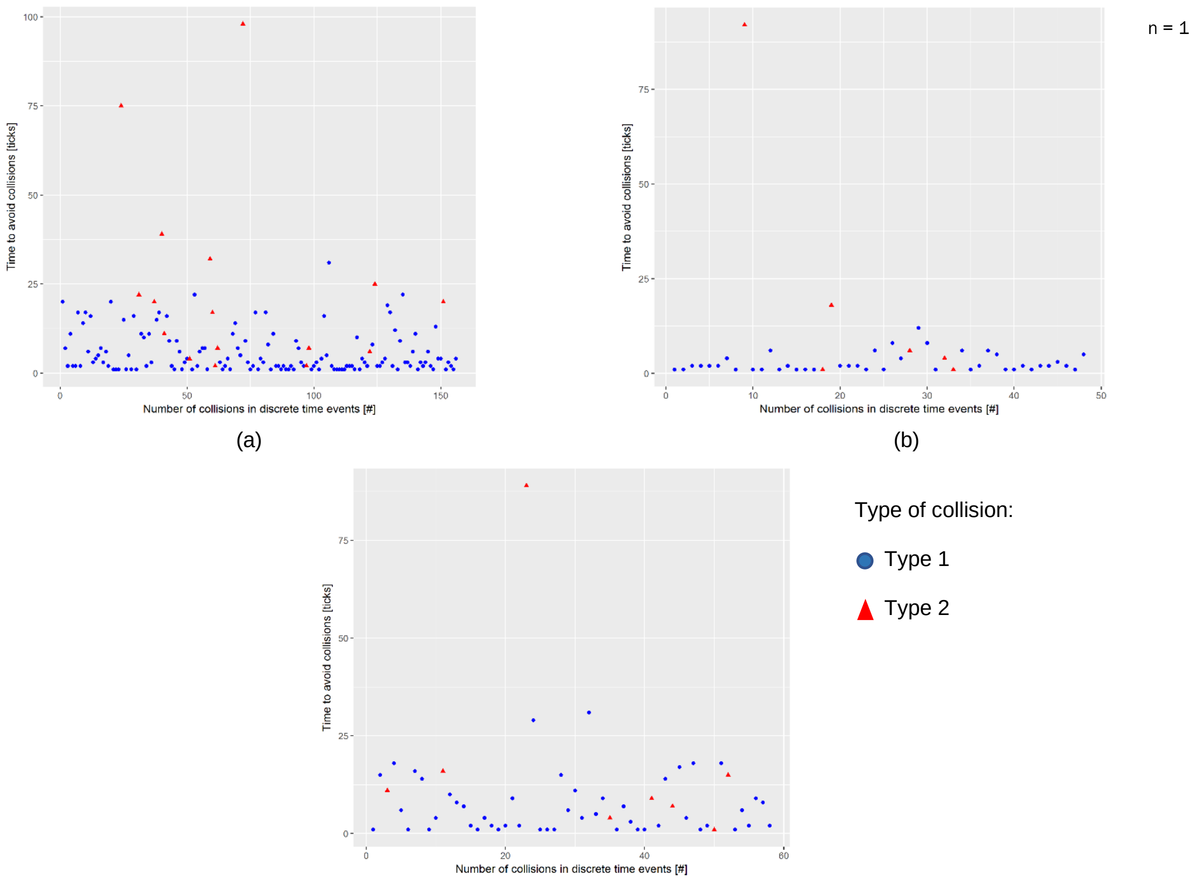
Figures 44a-44b: Heat-Map in a scenario with 100% light and low volume parcels (a) and 100% heavy and high volume parcels (b) with 100 robots

Comparing Figure 44a with Figure 44b, it can be immediately noticed that the coverage of space and concentration of robots is higher in the scenario with only light and low volume parcels (Fig. 44a), resulting in higher level of congestion in comparison to a scenario with only heavy and high volume parcels (Fig. 44b). This confirms that cooperative transport has a positive impact on the level of congestion inside the transport field. Therefore, we can also conclude that scenarios with cooperative transport do not demand larger workspaces compared to scenarios without heavy and high volume loads.

Congestion is also a suitable indicator for analyzing the safety of the system. Certainly, having more congestion in the transport field and, consequently more interferences between robots, increases the probability of collisions between robots. Therefore, we can conclude that Highway and Two-Layered guarantee higher safety compared to Mixed Traffic.

To corroborate these inferences, we can assess the results obtained in these scenarios on another performance indicator, being conditional congestion. This performance indicator indicates the time spent by robots to avoid collisions, when robots have a speed lower than their maximum speed. In our system, we can differentiate two types of collisions that robots encounter along their motion. The first type of collision implies the avoidance of robots without parcels or with light and low volume parcels. The second type of collision occurs when robots need to avoid formations of robots, i.e. robots transporting heavy and high volume parcels.
Figures 45a-45b-45c display the time spent by an individual robot during a run to avoid potential type 1 and 2 collisions. Since the maximum collision time may differ broadly from the average collision time, we have chosen to exhibit the results for an individual robot in a simulation run. The goal is to understand what types of collisions require longer time to be avoided. This time is measured in a scenario with 150 robots and 90% heavy and high volume parcels, in all traffic configurations.

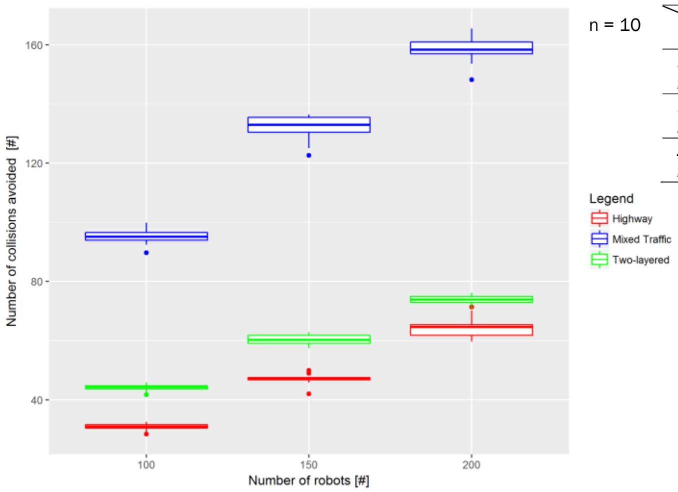
As can be observed, robots are generally able to resolve potential collisions in relatively short time intervals. However, type 2 collisions, i.e. potential collisions with formations of robots, might necessitate longer times to be avoided. These types of collision are indeed more troublesome, since robots need to dodge formations of robots travelling at reduced speed. The collision avoidance procedure becomes even more problematic when a single robot moves behind a formation without overtaking it. In these events, robots are forced to travel at reduced speed until the destinations of parcels, when the formation is dissolved. In Figures 45, the maximum collision time, corresponding to a type 2 collision avoidance, is equal to approximately 25 s (100 ticks). As seen in the simulation results, sometimes type 1 collisions might also demand long time to avoid, particularly when a robot has to avoid simultaneously multiple other single robots with conflicting trajectories.



(c)

Figures 45a-45b-45c: Time spent by robots during type 1 and type 2 collisions in (a) Mixed Traffic, (b) Highway and (c) Two-layered, with 150 robots and 90% heavy and high volume parcels

Furthermore, these figures confirm the conclusion inferred earlier concerning the low degree of safety guaranteed by Mixed Traffic compared to Highway and Two-layered. From these figures, it is already visible the higher number of collisions observed in Mixed Traffic, in comparison to Highway and Two-layered. Figure 46 displays the average number of collisions in the three diverse traffic configurations. It is apparent how the average number of interferences grows with increasing number of robots. Moreover, it is also evident that in Mixed Traffic robots need to avoid a higher number of collisions in comparison to Highway and Two-layered.



n = 10

	Mixed-Highway	Mixed-2Layered	2Layered-HIghway
100 robots	+ 209%	+116%	+30%
150 robots	+181%	+119%	+22%
200 robots	+145%	+116%	+12%

Figure 46: Average number of collisions in Mixed traffic (blue), Highway (red) and Two-layered (green) with 100, 150 and 200 robots and 90% heavy and high-volume parcels

As can be observed, the number of collisions in Mixed Traffic is extensively larger than the number of collisions avoided in Two-layered and Highway. With 100 robots, the number of collisions avoided in Mixed Traffic is +209% and +116% compared to Highway and Two-layered, respectively. In addition, it is also clear that Highway provides the best results in terms of average number of collisions, in every scenario. These results corroborate the outcome of Figures 42 and 43.

Table 14. Average collision time

	Mixed	Highway	2Layered
100 robots	7.66 t	4.7 t	5.68 t
150 robots	7.96 t	5.32 t	5.93 t
200 robots	8.45 t	5.78 t	6.34 t

n = 10

Furthermore, we are now able to calculate the conditional congestion, i.e. the average time spent by robots to avoid collisions, which corresponds to the product of the average collision time (table 14) and the average number of collisions (see Chapter 4, paragraph 4.4). Table 14 shows the average collision time for 100, 150 and 200 robots in the three traffic configurations, obtained after running 10 simulations for each scenario. Therefore, the average time spent to avoid all collisions in Mixed Traffic (*conditional congestion*) is equal to 182.13 s, 262.16 s and 334.4 s with 100, 150 and 200 robots respectively. In Highway, the conditional congestion is equal to 36.15 s, 62.4 s and 93.37 s with 100, 150 and 200 robots respectively. In Two-layered, the conditional congestion is equal to 62.58 s, 89.35 s and 116.35 s with 100, 150 and 200 robots respectively. Concluding, we can notice that the average time spent by robots to avoid collisions in half hour in Mixed Traffic is extensively longer in comparison to the time spent in Highway and Two-layered. This conclusion was expected considering the larger number of collisions robots need to avoid in this traffic configuration related to the other two traffic configurations. Whereby, we can conclude that separating the traffic in the transport field produces higher safety and time saving for collision avoidance.

5.3.3 Impact of cooperative transport on fault tolerance

In Chapter 2, we have defined fault tolerance as the ability of a system to keep operating even in the presence of failure of one (or more) of its elements. By developing disruptive experimental designs, we want to evaluate the robustness of the system. In particular, we desire to assess the impact of cooperative transport on the system fault tolerance.

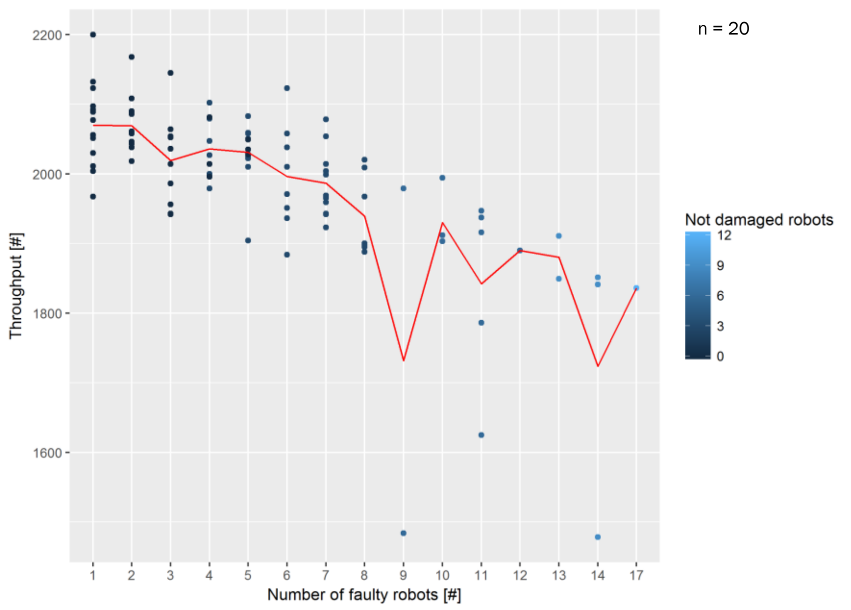


Figure 47: Results from experimental design four – robot failures

To do so, we have developed scenarios in Mixed Traffic with 150 robots and 90% light and low volume parcels (10% heavy and high volume parcels). As inferred in paragraph 5.3.2, Mixed Traffic features higher degree of congestion compared to the other two traffic approaches, and it is therefore more inclined to disruptive events. Furthermore, in paragraph 5.3.1, we have observed that the performance of robots in Mixed Traffic and with 90% light and low volume parcels are preferable with number of robots below 150, while the line of performance drops with 200 robots.

Therefore, we have decided to implement these scenarios with an optimal robot performance, meaning with 150 robots. In addition, 10% heavy and high volume parcel is a suitable percentage to show the impact of cooperative transport on fault tolerance.

In the fourth experimental design, we have limited the number of damaged robots from 1 to 5. This means that during a predefined time interval that we set initially, up to 5 robots stop functioning and remain into the transport field until the end of the shift, interfering with the motion of other robots. Considering that we want to evaluate the effect of disruption on system effectiveness, the temporal interval in which robots fail must be determined ex-ante. Spatial constraints are also introduced in this scenario, with robots having the possibility to fail merely in the transport field.

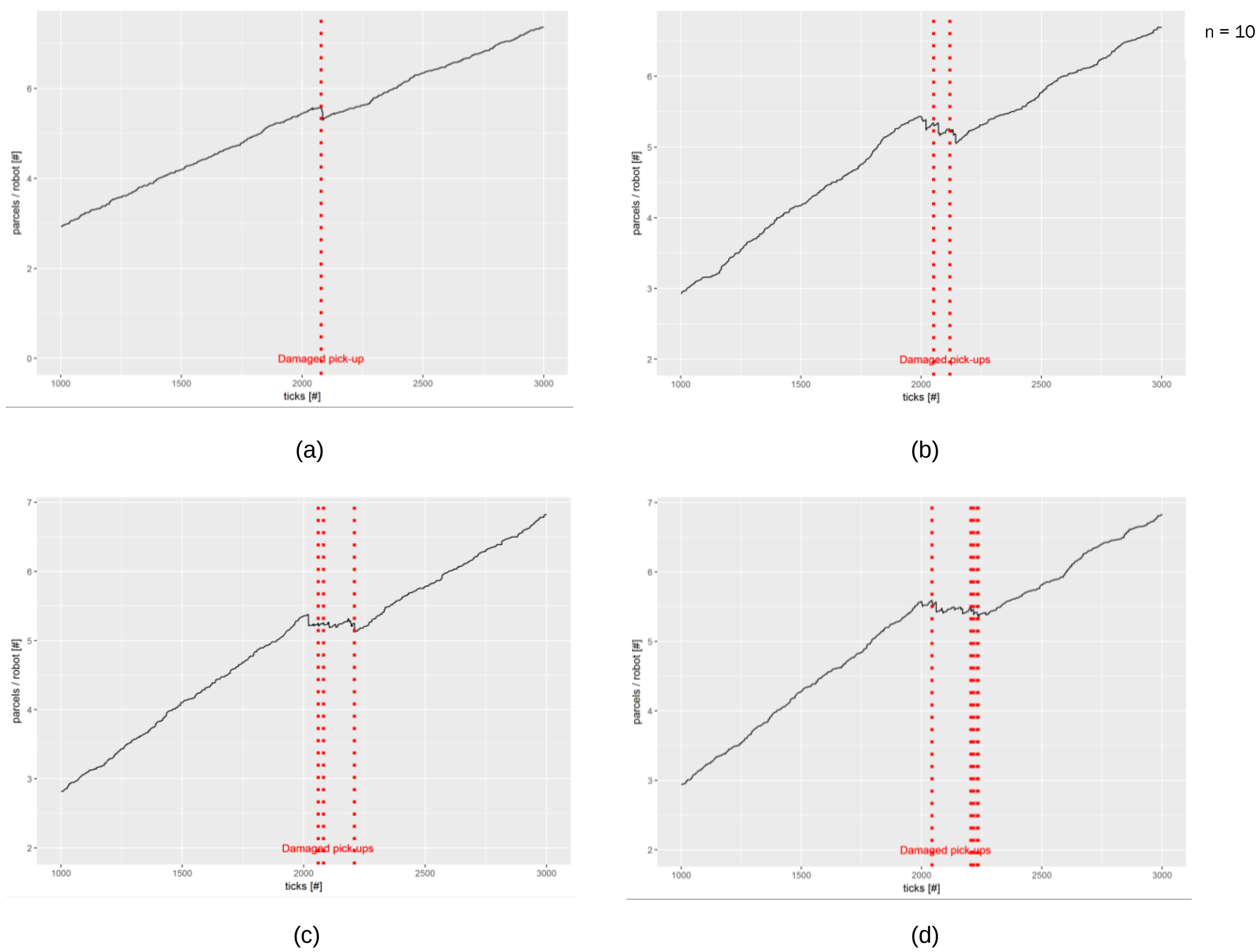
Under these circumstances, the impact of cooperative transport is easy to predict. In fact, when a robot fails in a formation, all the other robots in the formation, although not damaged, are unable to move. This results in more failures than anticipated. For instance, when we have one single failure, the number of idle robots can be up to four if the failure occurs in a formation; with 2 failures, we can instead have 2, 5 or 8 idle robots; with 3 failures, we can then have 3, 6, 9 or 12 idle robots, and so forth. In rare cases, robots can fail within the same formation thus changing the number of total failures.

The results from this experimental design are shown in the scatterplot in Figure 47. In this plot, on the x-axis we have the number of faulty robots, which includes damaged robots and not damaged robots (i.e. robots that are stuck because one other robot has failed in a formation), and on the y-axis we have the throughput. As it can be observed, the maximum number of faulty robots are 17, which means 12 failures more than the maximum number of failures set initially. In the plot, we have marked not damaged robots with a different gradient of color, to highlight the impact of cooperative transport. Comparing the throughput in standard conditions, without failures, with the throughput in the scatterplot, we can see how the red line, which indicates the mean, gradually decreases with the number of failures. With one and two failures, the mean throughput is 1.36% less (28 parcels less) in relation to the mean throughput in normal conditions. When the number of failures increases, with up to 4 failures, the mean throughput decreases by around 3% (around 60 parcels less). After 5 failures, the mean throughput drops below 2000 parcels in half an hour. The effect of cooperative transport on system effectiveness is explicit in this plot, with the mean of throughput dropping vigorously in just half hour. From this plot, it can be also observed that the probability that robots fail in formations is elevated, already with 10% heavy and high volume parcels. Indeed, we have found more failures than initially set in 40 out of 100 measurements (with 20 measurements per scenario). Therefore, we can conclude that cooperative transport has a strong negative impact on system fault tolerance.

In the fifth experimental design, we have loosen the spatial constraints for failing robots, with robots now having the possibility to fail inside pick-up buffers. When these events occur, pick-up buffers with inside failed robot(s) are declared impracticable. Therefore, robots can no longer access said pick-up buffers, and the ones moving towards those buffers need to recalculate a new destination to collect parcels. Therefore, the number of faulty pick-up buffers is dependent upon the spatial locations robots fail; for this reason, it can vary across the various simulation runs. In this research, we have limited our analysis to simulation runs with maximum four faulty pick-up buffers. Unlike the results presented earlier, in this experimental design, we have to examine results for single simulation runs, because results can vary significantly according to the number of faulty pick-up buffers. Furthermore, in this scenario, we do not focus on the repercussion of failures of robots and pick-up buffers on throughput. Instead, our interest is to examine the effect of failures of pick-up buffers on robot performance on the exact instants these failures take place. By doing so, we want to figure out how much time robots need to re-organize themselves before guaranteeing profitable performance again.

Therefore, the results on Figures 48a-48b-48c-48d display the impact of failures of pick-up buffers, highlighted with vertical red dotted lines, on robot performance at the exact failure instants, for single simulation runs. As it can be viewed in these figures, robot performance (i.e. parcels / robot) increases at the same rate until failure events. When one pick-up buffer fails (Fig. 48a), a small drop can be noticed, caused by the reduction of parcel supplies. It can be also deduced that robots need very little time to reorganize themselves in the eventuality of buffer failures. With the increase of the number of failed pick-up buffers, it can be noticed that robot performance requires more time (ticks) to continue increasing. Apparently, this corresponds to the time robots need to reconfigure themselves before keep functioning as usual.
Interestingly, when more than one pick-up buffers fail at different instances, robots require longer time to reposition across the supplying stations. However, if more than one pick-up buffers (Fig. 48d) fail at the same time, the reconfiguration time is approximately the same as when a single buffer fails. Additionally, the performance of robots until 3000 ticks (12.5 minutes) seems not to be affected by the increased

number of failed pick-up buffers. In fact, in all four simulation runs, at 3000 ticks robots have picked up around 6.7 parcels. However, a more resolute effect on robot performance is likely to be seen in the long run.



Figures 48a-b-c-d: Results from experimental design five with one (a), two (b), three (c) and four (d) failed pick-up buffers

It is important to remark that we are not trying to infer conclusions on whether the failure of pick-up buffers can have a stronger or softer impact on system effectiveness in comparison to the failures of robots. In order to do so, these two variables should have been separated in different experimental designs. Using this experimental design, we are able to see the robustness of the system, which is able to reconfigure in the events of other disruptive situations.

Ultimately, in the final experimental design, an assistance mechanism is implemented to address the impact of cooperative transport on fault tolerance, observed in Figure 49. This assistance mechanism consists of other robots placed outside the transport field, which intervene every time a robot fails. The assistance mechanism involves few elementary processes:

- When a robot fails, it communicates with one assisting robot. Just like for robot-parcel assignment, also in this case the robot-to-assistant is a 1-to-1 assignment, meaning that one assisting robot can be assigned to only one failed robot and vice versa.
- Once the message is arrived, the assisting robot moves into the transport field to help the failed robots. The assisting robot is not in charge of fixing the failed robots, but they only have to ensure that parcels on failed robots are delivered to the appropriate containers and that these robots are taken out from the field in order to not interfere with the motion of other robots.
- Therefore, once reached the position of the failed robots, the assisting robots check whether these robots have parcels with them or not. If they have a parcel, the assisting robots pick up the failed robots together with their parcels and transport them to the destination of the parcels.
- When the destination of the parcel is reached, the parcel is placed onto the right container. At this point, the assisting robots transport the failed robots outside the transport field for maintenance. However, the failed robots can no longer enter the transport field during the considered shift time.

It is important to notice that the scope of the assisting robots is (1) to deliver the parcels to appropriate destinations, (2) to eliminate interferences into the field between failed and not failed robots and (3) to eliminate the impact of cooperative transport on system fault tolerance. As a matter of fact, with regard to the formations of robots, the assisting robots only remove the damaged robots and transport them together with the parcels first to destinations and then outside the field. The other robots that were unable to move, as a consequence of the failure of a robot in formaiton, after the assistance, can again carry out their sorting operations. The outcome of the assistance mechanism is explicit when looking at the scatterplot in Figure 49 and compare it with the results obtained in Figure 47. In this plot, we can notice that the throughput declines, as a result of the increasing number of damaged robots. However, thanks to the assistance mechanism the impact of cooperative transport vanishes and we have no longer higher number of faulty robots than what initially defined. Therefore, in this plot the maximum number of faulty robots is exactly 5, while without assistance mechanism this number could arrive up to 20, as earlier explained.

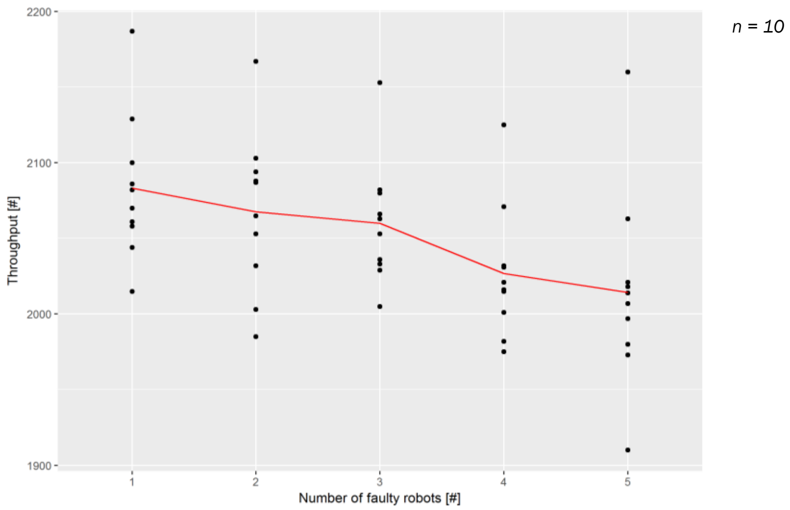


Figure 49: Results from experimental design six - robot assistance mechanism

Interestingly, the mean of throughput (red line) is approximately the same as the results obtained in the fourth experimental design, when the number of faulty robots is below 5. We can conclude that the robustness of the system is high also when the failed robots remain into the transport field. Moreover, the robustness of the system can be further increased by introducing an assistance mechanism as the one suggested in this research. It is important to highlight that the assisting robots are not identical to the sorting robots, as these have higher hardware capabilities compared to the sorting robots. The assisting robots can also be normal forklifts, with human operators transporting failed robots into and outside the transport field.

5.4 Comparison with Traditional Sorting Systems

In Chapter 2, we have described the strengths and weaknesses of traditional sorting systems. In Table 2, we have attributed to each property of conveyor systems a qualitative score. Accordingly, we have identified the main weaknesses of these systems, being volume flexibility, expansion flexibility, fault tolerance, utilization rate and reusability; and the main strengths, being throughput and reliability (see Chapter 2 for definitions of these characteristics). In this paragraph, we want to compare traditional sorting systems with the MRS designed in NetLogo.

By virtue of the high scalability of the developed MRS, this system is able to operate effectively with larger or smaller number of robots. This provides the new sorting system with high *Volume Flexibility*, i.e. the ability to operate profitably at different volume rates. Using this system, postal operators would be able to modify the number of robots according to the monthly, weekly, daily or even hourly demand. For instance, if the demand at instance x is equal to 1000 and at instance x + 1 is equal to 2000, postal operators can decide to introduce additional robots to satisfy the extra demand. Moreover, postal operators can also decide to increase / decrease the number of robots based on the different percentages of light-low / heavy-high volume parcels. Instead, conveyor systems are built with a maximum predefined throughput capacity and cannot operate at a volume rate larger than initially set. This forces postal companies to build machines according to the maximum historical demand. Therefore, these fixed machines guarantees low volume flexibility compared to multi-robot sorting systems.

Another weakness of conveyor system is low *Expansion Flexibility*. Unlike conveyors, the new parcel-sorting systems can be changed modularly in a warehouse. This means that postal companies can resize their sorting centers without need to add or remove equipment parts from sorting systems. In this way, postal operators can react to changing circumstances easily and readily. Beyond being adjustable, this system is also *Reusable*. Unlike conveyor systems that are built for specific layout of sorting centers, or that even force postal operators to construct sorting centers with specific layouts, this multi-robot parcel sorting system can be used in any configuration. Although this strength of the MRS has not been shown in this research, with the use of a single fixed layout, we are confident that this system can work under changing layouts. Indeed, the algorithms employed for the implementation of this system combine reactive and deliberative control methods. The localization and mapping algorithm is fully decentralized and allows postal operators to use the system to map any layout. The path planning and collision avoidance algorithms (phi* or basic theta* plus swarm intelligence techniques) eliminate the dependence on fixed infrastructure. The min-max heuristics uses a hybrid approach with feedback between optimizer and robots. When used in other configurations, only the maximum distances calculated to minimize distance travelled of robots should be revised. The leader-follower algorithm is a centralized (deliberative) algorithm, but it is reusable in any other circumstances, with more or less robots in formations. The construction of the fixed path outside the transport field (i.e. Highway) for the separation of loaded and unloaded robots should be revised according to the layout of the sorting hub. The Two-Layered is the only traffic configuration that requires large improvements of layouts in case UGVs are used. Therefore, in comparison to traditional sorting systems, we can consider the new sorting system reusable.

The utilization rate of conveyor systems decreases in certain periods of the year, when the demand is lower in comparison to the maximum throughput capacity. As already argued, the new sorting system has superior scalability, which allows postal operators to use less or more robots depending on the demand. Consequently, this enables the MRS to achieve high utilization rate. Furthermore, considering the high flexibility/modularity of this system that can be changed according to the situation, the utilization rate can be further improved by introducing structural changes in correspondence of the changing demand. In fact, the utilization rate can be also improved by adding or removing pick-up buffers or drop-off buffers to reduce the distance travelled by robots. The introduction or removal of buffers from the field is even possible within the single shift and robots are able to reorganize themselves in a very short timeframe, as shown by the results of Figures 48.

In addition, conveyor systems have low fault tolerance, as they constitute a single point of failure. It is important to underline that fault tolerance and reliability are two different elements. The former concerns with the ability of systems to operate in presence of failures, while the latter concerns with the ability of systems to work for long periods without failures. Earlier, we have talked extensively about the fault tolerance of the MRS. Considering the results obtained, we can infer that the MRS shows off high fault tolerance. Pertaining to the reliability of the MRS, we cannot estimate whether this is low, average or high. In order to do so, we would need to calculate the probability of failure for each robot and multiply it for all the robots used. We also know that the failure rate would look like a bathtub curve consisting of three periods, namely infant mortality, constant failures and wear out failures (G.A. Klutke et al., 2003). The objective of maintenance planners should then be to keep the failure rate as much as possible within the middle phase. However, without extra information about the maintenance or failure rate of robots, we cannot infer whether this system is more or less reliable compared to conventional sorting devices.

Besides their high reliability, conveyor systems have the benefits of high throughput capacities. As can be seen in Table 15, conveyors can achieve up to 12,000 parcels in a single hour with using 50 sorting directions, i.e. drop-off buffers, 11 chutes placed at the end of the sorting directions and in an area of 3377 square meters (data provided by PostNL).

From the results of our experiments with only light and low volume parcels, the MRS can achieve in Mixed Traffic around 4000 parcels in half hour (around 8000 per hour), in an area of 3750 m^2, using 200 robots. To achieve the same performance, the designed system would require around 300 robots, in an area bigger than the one considered. However, it must be stated that throughput also depends on the number of supplying platforms used by the sorting systems. Increasing the supplies would certainly increase the performance. Moreover, it can be stated that the conveyor systems extends vertically other than horizontally, therefore the traditional sorting centers have large heights. When this area is divided in different floors, multiple MRS can be implemented on the different floors, thus achieving and even exceeding conveyors' performance.

Table 15. Conveyors data performance, data provided by PostNL

Max throughput [hour]	Sorting directions [#]	Chutes [#]	Total area required [m^2]
12,000	50	11	3377.04
10,000	50	9	3089.04
8,000	50	8	2913.04
5,000	50	8	2657.04
2,000	50	8	2465.04

Finally, it is relevant for this thesis to underline that conveyor systems are only able to move and sort parcels of specific dimensions (see Chapter 2 for maximum dimensions of parcels in PostNL). Therefore, outsized parcels cannot be sort automatically, but need to be sort manually. In the design system, using cooperative behaviors, robots are able to transport and sort parcels of any size.

Conclusions

In this Chapter, first we have presented three different traffic design alternatives, being Mixed Traffic, Highway and Two-Layered. The last two traffic configurations can be employed to separate loaded and unloaded robots on the same plane or on two planes; while, in the first traffic configuration, both entities move inside the transport field. Subsequently, we have described the experimental designs that we have developed in order to deduce conclusions with regard to the impact of cooperative transport on respectively system effectiveness, congestion and fault tolerance.

Altogether, from the analysis of the results from the first three experimental designs and in consideration of the impact of cooperative transport on system effectiveness, we have formulated the following conclusions:

- With increasing percentages of heavy and high volume parcels, the difference in throughput between Mixed Traffic and Two-layered becomes larger, while the difference between Mixed Traffic and Highway becomes smaller. Thus, cooperative transport influences largely Mixed Traffic.
- We have also inferred that cooperative transport strongly reduces the throughput, because it requires four robots to transport a single parcel. Furthermore, we have noticed that in a scenario with higher percentage of heavy and high volume parcels, the system necessitates higher number of robots to achieve comparable results in terms of throughput. This is caused by the longer intervals of time robots wait at pick-up buffers before building formations of robots. This is further confirmed by the robot performance statistics, which show that higher performances are achieved with higher number of robots in scenarios with heavy and high volume parcels.
- Additionally, these statistics demonstrate that Mixed Traffic tolerates poorly over 150 robots, while Highway and Two-Layered favor high number of robots. In the last two configurations, the lines representing robot performance converge as a result of declining line with 100 robots and increasing lines with 150 and 200 robots.
- Service time and percentage distance travelled idle are two main drivers for throughput. We have observed that in Mixed Traffic the service time is higher than in the other traffic design alternatives, while the distance travelled idle is lower than in Highway and equal to the distance travelled idle in Two-Layered. Therefore, overall Mixed Traffic offers strong results with regard to distance travelled idle, but poor results with regard to service time in comparison to the other two configurations. With Highway, we achieve the opposite results, meaning that this traffic strategy provides low service time, but very high percentage of distance travelled idle. In comparison, Two-Layered guarantees low service time and percentage of distance travelled idle.
- We have observed that cooperative transport influences service time, considering that robots in formation travel at half speed as the robots without parcels or with small parcels. Moreover, cooperative transport slightly increases the service time for light and low volume parcels, since non-cooperative robots need in some events to travel behind robots in formations, which slows down their motion, and need more time to avoid collisions with them. In addition, it can be concluded that cooperative transport does not have any impact on the percentage distance travelled idle.

In consideration of the results from the first three experimental designs, but this time regarding the impact of cooperative transport on congestion, we have inferred that:

- Mixed Traffic shows the highest level of congestion, due to the absence of separation of different entities. In this scenario with 100% light and low volume parcels, the level of congestion almost doubles from 100 to 200 robots. This entails that Mixed Traffic tolerates to a lower degree higher number of robots compared to Highway and Two-Layered.
- In a scenario with 100% light and low volume parcels, the level of congestion has a lower impact on throughput than in a scenario with 10% heavy and high volume parcels. This by reason of the larger number of robots introduced in the system compared to the number of pick-up buffers. For instance, in a scenario with 100 robots, there are 5 robots per pick-up buffers. This entails that in Two-Layered robots wait a bit longer before entering these buffers compared to Mixed Traffic, where due to the high level of congestion robots arrive a bit later at pick-up buffers. Therefore, the waiting time of parcels in pick-up buffers does not change when comparing these two traffic configurations, as can be seen from the results in Appendix O. However, when the percentage of heavy and high volume parcels increases, congestion has a larger impact on throughput, because pick-up buffers request a higher number of robots. Therefore, arriving with a certain delay at pick-up buffers will have a stronger influence in these scenarios.
- Cooperative transport reduces the level of congestion due to a relative lower coverage/concentration of robots in the space. This entails that in scenarios with high percentage of heavy and high volume loads, there is no need to increase the workspace inside a sorting hub.
- Congestion is a good indicator of safety; therefore, we can consider systems working in Mixed Traffic less safe than systems working in Highway or Two-Layered. In particular, Highway is among the others the safest option, due to the minimization of side-collisions and the predictable behavior of robots.
- Along their motion, robots may encounter two types of collision, one with robots without parcels or with small parcels and another one with formations of robots transporting big parcels. As shown in Figures 45, the latter might require very long time to be avoided.
- Figures 46 show that in Mixed traffic robots need to avoid an extensively larger number of collisions compared to Highway and Two-layered. These results corroborate the inference made earlier, according to which Mixed Traffic

produces higher level of congestion in comparison to the other two configurations.

- Overall, it was inferred that in Mixed Traffic, robots spend 182.13 s, 262.16 s and 334.4 s to avoid collisions with 100, 150 and 200 robots respectively. In Highway, robots spend 36.15 s, 62.4 s and 93.37 s to avoid collision with 100, 150 and 200 robots respectively. In Two-layered, robots spend 62.58 s, 89.35 s and 116.35 s to avoid collisions with 100, 150 and 200 robots respectively. Concluding, separating the traffic in the transport field can produce higher safety and time saving to avoid collisions.

By developing and analyzing results from three experimental designs that considered disruptive scenarios, we have deduce the following conclusions:

- The impact of cooperative transport on system fault tolerance was reported in the scatterplot (Figure 47). When a robot fails in a formation, it forces the other robots to stop functioning. This entails that even when a single robot is demoted in a formation, in reality four robots are unable to operate.
- Looking at the scatterplots in Figures 47 and 49, we have observed that the system shows off high robustness, considering that even in the eventuality of 5 failures, the mean of throughput only reduces by 5% compared to the standard output, in the considered timeframe.
- In consideration of experimental design five, where the failure of pick-up buffers is analyzed, we have inferred that robots need relatively short time to recalculate a new parcel-supplying platform and reposition themselves across the various pick-up buffers. Moreover, when more pick-up buffers fail at the same instant, the time to recalculate their position is not different from the event of a single buffer. In the considered timeframe, after the reconfiguration time, the performance of robots continue to grow steadily.
- A solution to tackle the problem brought about by cooperative transport on system failure is presented. This strategy considers the use of heterogeneous robots placed outside the field, helping remove failed robots and deliver parcels to appropriate destinations. Thanks to this solution, the impact of cooperative transport on system fault tolerance is eliminated, as shown in Figure 49. Indeed, robots that are stuck in a formation due to the collapse of one teammate, after the removal of the damaged robots, can keep operating profitably.

Finally, we have compared our designed parcel-sorting MRS with conveyor systems. In comparison to these systems, the new systems offer (1) higher volume and (2) expansion flexibility, (3) reusability, (3) higher fault tolerance, and (4) higher utilization rate. It is difficult to compare this system with traditional sorters in terms of reliability, for which we would need data regarding the failure rate of robots. Concerning with throughput, we have seen that these systems would require more robots and workspace to achieve the maximum throughput of conveyor systems. However, it can also be concluded that this system has the potential to achieve the same performance with a profitable number of supplies or using elevated spaces. The other advantage of the MRS is that it is able to handle parcels of any size, thanks to the cooperative behaviors of robots.

In the next Chapter, we answer the sub-questions and the main research questions; describe the scientific and societal relevance of this research; provide recommendations for future work, both for the research in the scientific field and for practical use of this system; and, describe the limitations of the choices made in this research.

6 | Conclusions

Introduction

This is the conclusive Chapter of my master thesis dissertation. In this Chapter, the main objective is to answer the research questions, presented in Chapter 1. Furthermore, we announce the main thesis contributions, both from a scientific and societal perspective; we reflect upon the limitations of the methods used in this project; and we provide recommendations to Prime Vision and researchers for future course of actions.

Therefore, this Chapter is structured as follows. In paragraph 6.1, we declare again the project goal and we answer the research questions. In particular, in sub-paragraphs 6.1.1 to 6.1.5, we answer the sub-questions, which taken together ease answering the main research question. The answer of the main research question is provided in sub-paragraph 6.1.6. In paragraph 6.2, we reflect upon the scientific and societal relevance of the results obtained in this thesis project. In paragraph 6.3, we discuss the differences between our multi-robot system and previously implemented multi-robot systems. In paragraph 6.4, we describe the most important limitations of this work, especially pertaining to the developed model. Finally, in paragraph 6.5, we suggest future plans of action for Prime Vision and for researchers that want to improve and expand the developed model.

6.1 Project goal and research questions

In recent years, the postal industry has experienced turbulent periods due to the strong impact of e-commerce on postal operations. E-commerce is considered as the catalyst factor of changing market dynamics, leading to strong fluctuation of volumes and of dimensions of parcels to sort. Postal operators want to address this problem without being obligated to purchase continuously new sorting centers and equipment, which is not a sustainable and future proof solution. In the Netherlands, PostNL is collaborating with Prime Vision on the development of a new sorting system that can provide higher flexibility, scalability, fault tolerance and comparable performance by using a multi-robot approach. Within this project, our responsibility was to develop a macro-model featuring the operations of multiple robots performing parcel-sorting operations in a warehouse. Another task was also to develop and simulate a solution for the transportation of heavy and high volume parcels. To this end, we have decided that homogenous robots should be able to switch dynamically their roles when facing different types of tasks, showing off both cooperative and non-cooperative behaviors. Furthermore, together with Prime Vision, we have defined the main goal of this project that is to explore the possible system effectiveness, congestion and fault tolerance effects of cooperative robots.

To achieve this research objective, we have acted in accordance to the DSRM methodology of K. Peffers et al. (2008). Additionally, while following the framework of this dissertation, we have investigated the research sub-questions, which facilitate and ease answering the main research question. Therefore, we first provide an answer for each sub-question and, subsequently, answer the main research question that instantiates the achievement of the goal for this project.

6.1.1 Answer to research sub-question 1

In this paragraph, we present an answer to the first sub-question, being: *"How can cooperative and non-cooperative transport be modelled and formalized within the same application?"*

From the study of the literature (Chapter 3), we have observed that there are several ways to coordinate the motion of robots, namely leader-follower, virtual structure, graph-based, artificial potential field and behavioral approach. In Chapter 4, we have explained that the choice was restricted to two approaches being the leader-follower approach or the virtual structure approach that can guarantee stability to the formation and to the parcels to transport. In the virtual structure approach, the formation of robots is considered as a single physical object and desired trajectories are assigned to the entire formation as a whole. As explained in L. Consolini et al. (2008), this approach requires large inter-robot communication bandwidth, given that robots exchange continuous feedbacks to avoid some members to leave the formation. Another disadvantage of the virtual structure approach, as seen in I. Mas and C. Kitts (2010), is its low scalability, since introducing additional robots to formations affects the structure / physics of the rigid body. Instead, leader-follower approaches are acknowledged for being scalable and simple. The level of communication is drastically reduced compared to the virtual structure, considering that the communication is unilateral leader-to-followers. In our system, followers have the objective to meet the speed, distance and steering angle (orientation) of leaders. However, as shown in our system, the leader exploits the data collected by followers to expand its vision radius and avoid collisions. Additionally, the computations required for path planning and collision avoidance are also reduced, since the leader only computes the best path to follow and determines the trajectories to use to avoid potential collisions. The disadvantage of both approaches is their centralized nature, which reduces their robustness.

Concerning with the allocation of robots to pick-up buffers, we have referred to the study of A. Farinelli et al. (2017) where a max-sum heuristics was used to allocate robots to tasks in such a way that maximize the throughput and minimize the travel time of robots. However, considering the types of tasks involved in our application, which includes both ST-SR-IA and ST-MR-IA, we have used a min-max heuristics, with the objective of minimizing the maximum waiting time, while keeping short the travelled distance of robots.

The researches of Z. Wang and M. Schwager (2016) and B. Hichri et al. (2016) were useful to understand how the joint effort of multiple robots can be used to transport cooperatively heavy and differently shaped loads. Furthermore, the research of L.E. Parker and F. Tang (2007) confirms the feasibility of using robots to perform ST-SR-IA and ST-MR-IA tasks within a single application. Nevertheless, as explained in Chapter 4, in comparison to that application, in this domain we are using homogenous robots (robots with identical capabilities) and without announcing tasks sequentially. Therefore, the accomplishment of highly cooperative and weakly cooperative tasks, within the same application with task uncertainty, represents one of the knowledge gaps for this dissertation. In order to perform both types of tasks within a single application, in our system, robots change dynamically their roles when facing the different classes of tasks. This entails that when robots need to handle weakly cooperative tasks, they decide to act individually; while, when confronting strongly cooperative tasks, one robot assumes the role of the leader and becomes an auctioneer, recruiting followers based on their status. Therefore, we have combined the leader-follower with an auction-based algorithm to build up formations of robots. Hence, we have modelled robots able to react instantaneously to the different tasks, assuming non-cooperative and cooperative behaviors based on the tasks to handle.

Furthermore, once the formations are built, leaders can assume different positions within the formations that give them the highest situational awareness. A similar technique was used in J.P. Desai et al. (2001), to alter the shape of formations when facing diverse obstacles. In this application, leaders can position themselves on the front left or front right, dependently on the destination of the parcels.

6.1.2 Answer to research sub-question 2

The second sub-question aims at defining the key performance indicators that are used in the experimental designs for the evaluation of the effectiveness and fault tolerance of the new parcel-sorting MRS. Accordingly, the research question is: *"How to quantify system effectiveness, congestion and fault tolerance with specific multi-robot performance indicators?"*

The answer to this sub-question can be traced in Chapter 4, paragraph 4.4. The performance indicators needed to quantify the KPIs of the new multi-robot parcel sorting system, being system effectiveness, congestion and fault tolerance, are:

- *Average utilization rate of robot*: average number of robots with loaded status;
- *Throughput*: total number of parcels on drop-off buffers (or containers). This can also be considered as the total number of tasks accomplished per unit of time;
- *Average robot performance*: average number of parcels per single robot. This indicator can be easily obtained dividing throughput by the total number of robots;
- *Utilization rate of pick-up buffers*: average number of robots per pick-up buffers, including robots in entry queues connected to buffers and robots inside the buffers.
- *Average waiting time of parcels*: average time waited by parcels at pick-up buffers before being picked-up by robots.
- *Average service time of parcels*: average time required by robots to transport parcels from pick-up to drop-off buffers.

- *Total time in system for parcels*: sum of the average waiting time of parcels and the average service time of parcels.
- *Average robot distance travelled idle*: average distance travelled by robots without parcels divided by total distance travelled by robots. This performance indicator gives insights into the percentage of distance travelled idle (without parcels) of robots.
- *Average congestion*: congestion here is computed as a function of speed. The average speed of robots is divided by the maximum speed of robots to gain insights into the number of interferences robot-to-robot. This quantity is summed up for all robots in the system and divided by the total number of robots to obtain the percentage level of congestion.
- *Conditional congestion*: time spent avoiding collisions, i.e. when speed of robots is lower than their maximum speed. This is measured as the product of average collision time for all robots and average number of collisions.

Most of these performance indicators can be employed in sorting centers to evaluate the performance of conventional sorters. In addition to these generic indicators, *average utilization rate of robots, average robot performance, average distance travelled idle, average congestion* and *conditional congestion* need to be integrated to evaluate comprehensively the performance of a multi-robot parcel sorting system.

6.1.3 Answer to research sub-question 3

The third sub-question of this dissertation considers the design of different traffic configurations to control the motion of robots in different ways. Accordingly, the sub-question is: *"What design alternatives can be adopted to control the traffic flow of robots inside a sorting hub?"*

The answer to this question can be found in Chapter 5, paragraph 5.1. Building on the approaches used in transportation to supervise traffic and increase safety of travelers, we have developed three traffic control strategies.

In the first traffic design alternative, named *Mixed Traffic*, robots with parcels and without parcels move in opposite trajectories sharing the same field (see Figure 33a). This traffic alternative is likely to have high number of interferences, but short distance travelled by robots. In order to reduce the distance travelled of robots in this configuration, we have improved the min-max heuristics used for resource allocation constraining the distance robots can travel to reach pick-up buffers. This distance optimization strategy also allows reducing the interferences among robots, which will follow a curved trajectory.

In the second traffic design alternative, named *Highway*, the motion of robots without parcels is separated from the motion of robot with parcels, on the same plane (see Figure 33b). In this configuration, robots travelled with parcels inside the transport field until the destination of parcels, while robots without parcels follow reference points outside this field. The advantage of this traffic alternative is the minimal number of interferences between loaded and unloaded robots, which only occur when unloaded robots reach the final reference points that steer them towards the pick-up area. To manage profitably these remaining interferences, robots with parcels are assigned with higher priority compared to robots without parcels. Other than guaranteeing adequate safety, this traffic alternative increases the predictability of the system, with the behaviors of robots becoming easier to understand and control. By contrast, this alternative is likely to lead to robots travelling long distances before returning to pick-up stations. It is important to mention that the maximum distance robots can travel to reach a pick-up buffer is calculated differently in this case, since robots select their pick-up buffer only once the last reference points are reached. The distance again aims to reduce the distance travelled by robots and the number of interferences between robots.

In the third traffic design alternative, named *Two-Layered*, the motion of robots without parcels is again separated from the motion of robots with parcels, but on two planes (Figure 33c). Specifically, robots without parcels travel on the second plane, while robots with parcels travel on first plane. The advantage of this traffic control strategy is again the reduction of the number of interferences between robots moving in opposite directions. Moreover, this specific traffic configuration also guarantees short distances travelled by robots, other than short service and return time. The maximum distance travelled by robots is calculated as in Mixed Traffic. It is relevant to notice that when using UAVs, the two layers are not physical floors, while when using UGVs, the two layers correspond to two different floors. Therefore, in the second case (using UGVs), this alternative would have the disadvantage of requiring larger workspaces.

These traffic design alternatives represent one of the input parameters for the experimental designs of this research.

6.1.4 Answer to research sub-question 4

The fourth sub-question follows the analysis of the results from the experimental designs that focus on the impact of cooperative transport on system effectiveness and congestion. Accordingly, the fourth sub-question is: *"What is the impact of cooperative transport on system effectiveness and congestion?"*

The answer to this research question can be found in Chapter 5, paragraphs 5.3.1 and 5.3.2. In these paragraphs, we have reflected upon the results of the first three experimental designs, listed in Table 9 of paragraph 5.2. From this analysis, we have made the following inferences regarding the impact of cooperative transport on system effectiveness:

- In a scenario with 100% light and low volume parcels, Mixed Traffic and Two-Layered achieve similar results. Furthermore, both traffic strategies outperform evidently Highway in this scenario. Using this system, we can achieve a maximum throughput of over 4000 parcels in half hour in Mixed Traffic and Two-Layered.
- Cooperative transport strongly reduces throughput, given that more resources are used to transport a single parcels. Among the designed alternatives, Mixed Traffic is the most negatively affected configuration. In fact, in a scenario with 10% heavy and high volume parcels, the mean of throughput reduces by 55% with 100 robots, 52% with 150 robots and 50% with 200 robots.
- Cooperative transport requires higher number of robots to achieve a level of throughput comparable to that obtained in a scenario with zero percent heavy and high volume parcels. This is due to the long time interval waited by robots at pick-up buffers to build formations of four robots. Therefore, the higher the number of robots, the higher the probability formations are built.
- With increasing percentages of heavy and high volume parcels, the difference in throughput between Mixed Traffic and Two-Layered becomes larger, while the difference between Mixed Traffic and Highway becomes smaller, particularly with high number of robots. While, large difference in throughput remains between Highway and Two-Layered.
- Considering the results of robot performance (parcels per robot calculated over time), the lines indicating the mean performance converge in Highway and Two-Layered, in a scenario with 10% heavy and high volume parcels. This indicates that the performance decreases with low number of robots, while increases with increasing number of robots. This confirms our hypothesis that

cooperative transport requires higher number of robots. These results have also demonstrated that in this scenario Mixed Traffic tolerates efficiently up to 150 robots.

- Cooperative transport does not influence considerably the service time of light and low volume parcels, but it adversely impacts the total average service time of parcels, in consideration of the lower speed travelled by robots in formation. In particular, the service time of light and low volume parcels is around 41-45 seconds in Mixed Traffic and around 39-41 seconds in Highway (same for Two-Layered). Instead, the service time of heavy and high volume parcels is around 93 seconds in Mixed Traffic and 82-85 seconds in Highway and Two-Layered.
- Cooperative transport does not influence the percentage distance travelled idle of robots. In every scenario, the percentage distance travelled of robots is around 45% in Mixed Traffic and Two-Layered, while it is around 58% in Highway. Therefore, in Highway, robots travel significantly more (+8%) without parcels than with parcels.

Concerning the impact of cooperative transport on congestion, we have made the following inferences:

- In every scenario, Mixed Traffic shows off the highest level of congestion, while Highway bears the lowest level of congestion. In comparison to the other two traffic design alternatives, the degree of congestion rises exponentially with increasing number of robots in Mixed Traffic, meaning that this solution tolerates poorly high number of robots.
- Congestion is a good indicator for the level of safety, as it indicates the amount of interferences in the area. Therefore, we can conclude that Mixed Traffic is the unsafest option, while Highway and Two-Layered can be both considered highly safe.
- Cooperative transport reduces the level of congestion due to the lower concentration of robots in the space. The implementation of heat-maps help understand this phenomenon (see Figures 46a-46b).
- Cooperative transport influences the impact that the level of congestion has on throughput. This entails that in a scenario with 100% light and low volume parcels, the level of congestion has low or zero impact on throughput. While this factor becomes gradually more influential with higher percentages of heavy and high volume parcels. The reason lies on the fact that pick-up buffers become more demanding, i.e. need to be fed with more robots, with increasing number of heavy and high volume parcels. In the scenarios with zero percent

big loads, robots wait more time before accessing the pick-up buffers compared to scenarios with heavy and high volume parcels.

- Along their motion, robots may encounter two types of collision, one with robots without parcels or with small parcels and another one with formations of robots transporting big parcels. As shown in Figures 45, the latter might require very long time to be avoided.
- Figures 46 show that in Mixed traffic robots need to avoid an extensively larger number of collisions compared to Highway and Two-layered. These results corroborate the inference made earlier, according to which Mixed Traffic produces higher level of congestion in comparison to the other two configurations.
- Overall, it was inferred that in Mixed Traffic, robots spend 182.13 s, 262.16 s and 334.4 s to avoid collisions with 100, 150 and 200 robots respectively. In Highway, robots spend 36.15 s, 62.4 s and 93.37 s to avoid collision with 100, 150 and 200 robots respectively. In Two-layered, robots spend 62.58 s, 89.35 s and 116.35 s to avoid collisions with 100, 150 and 200 robots respectively. Concluding, separating the traffic in the transport field can produce higher safety and time saving to avoid collisions.

6.1.5 Answer to research sub-question 5

The fifth sub-question focuses on the impact of cooperative transport on the robustness of the multi-robot parcel sorting system, meaning the ability of system to keep working effectively even in the presence of failures (robots or buffers). Accordingly, the fifth sub-question is: *"What is the impact of cooperative transport on fault tolerance?"*

The answer to this research question can be again found in Chapter 5, paragraph 5.3.3. In this paragraph, we have reflected upon the results of the last three experimental designs (4, 5 and 6), listed in Table 9 of paragraph 5.2. From this analysis, we have made the following inferences regarding the impact of cooperative transport on fault tolerance:

- The multi-robot parcel sorting system offers high robustness even in the event of five robot failures (see Figures 47 and 49). Considering the results obtained in the fourth and fifth experimental design, the mean throughput decreases by merely 5% when five robots fail in the transport field (in half hour).

- Cooperative transport has a strong negative impact on system fault tolerance, given that perturbation of a robot in a formation brings about a collapse for the whole members of said formations. Consequently, one failure in a formation corresponds to 4 failures, two failures in two separate formations correspond to 12 failures and so forth. Therefore, cooperative transport decreases the robustness of the system and can lead to a resolute decrease of system effectiveness.
- To address the drop of robustness caused by cooperative transport, an assistance mechanism should be introduced. This assistance mechanism considers the use of highly capable robots / machines (e.g. forklifts), placed outside the transport field, helping remove failed robots from the field and deliver parcels to appropriate destinations. By means of the assistance mechanism, the number of interferences robots – damaged robots decreases and the disruptive effect of cooperative transport on system fault tolerance vanishes. In fact, after removing damaged robot from formations, the trapped robots will continue operating effectively.
- The results from experimental design five have demonstrated that robots require short time to recalculate a new pick-up buffer and reposition across the pick-up buffers. Moreover, these results have shown that the recalculation time does not differ when multiple pick-up buffers fail at very short instances compared to one single failure. After the reconfiguration, robots keep working profitably again.

6.1.6 Answer to main research question

Taken together, the answers of the research sub-questions lead to an answer of the main research question:

"What does an effective and robust multi-robot parcel sorting system design look like in which robots behave in a cooperative and non-cooperative manner?"

First, we have explained that the multi-robot parcel-sorting system should be able to operate ST-SR-IA (weakly cooperative) and ST-MR-IA (strongly cooperative) tasks, within the same application. To do so, robots switch dynamically their roles/behaviors when facing the different types of parcels (tasks). For the assignments of tasks to robots we have implemented a min-max heuristics, which intends to minimize the maximum waiting time of parcels at pick-up buffers while keeping short the distance travelled by robots. When a robot is assigned with ST-MR-IA tasks, it switches its role into a leader, starting searching for followers. The combination of leader-follower and auction-based algorithm guarantees simplicity, stability and scalability. Robots assigned with ST-SR-IA tasks behaves in a non-cooperative manner, transporting parcels to destinations individually.

Different design strategies to control the traffic flow of robots inside the sorting center are proposed, namely Mixed Traffic, Highway and Two-Layered. From the results of the experimental designs, we have inferred that Two-Layered offers higher system effectiveness (throughput, distance travelled and service time) compared to Mixed Traffic and Highway in any scenario. When UAVs are used in place of UGVs, there is no need to have larger physical workspace with this solution. Mixed Traffic offers comparable throughput in a scenario with 100% light and low volume parcels. However, cooperative transport has the strongest negative effect on throughput in this traffic configuration. Another conclusion was that Mixed Traffic tolerates poorly 200 robots with increasing percentages of heavy and high volume parcels. This is to attribute to the level of congestion in this configuration, which adversely affects throughput with increasing amount of heavy and high volume parcels. In comparison, Highway has the advantage of having shorter service time, but robots in this configuration travel over 10% more idle (without parcels). Nevertheless, from the results it was apparent that the difference in throughput between Mixed Traffic and Highway reduces with increasing percentages of heavy and high volume parcels and with increasing number of robots. The results from the experimental designs made it clear that in order to make the system effective in scenarios with high number of heavy and high volume parcels, a higher number of robots needs to be introduced. Moreover, the results from the analyses prove that the implementation of cooperative transport does not lead to larger workspace requirements, given that the concentration of robots in the space is diminished.

The robustness (high fault tolerance) of the multi-robot parcel-sorting system is demonstrated in the analysis of the results from the disruptive experimental designs. Indeed, robots operate with high performance even in presence of failures. In the considered timeframe and after the failure of five robots, the mean throughput was still over 2000 parcels (-5% in comparison to the standard throughput). To further increase system robustness, an assistance mechanism should be implemented, which aims at removing robots from the transport field, delivering parcels on failed robots at destinations and, especially, disengage trapped (not damaged) robots from the formations. By doing so, the negative effect of cooperative transport on system fault tolerance is eliminated.

6.2 Relevance of thesis contributions

This research has a socio-technical facet, with contributions provided to both the academic and societal fields.

6.2.1 Scientific Relevance

The main scientific contribution of this research is the elaboration of an algorithm for the fulfillment of weakly cooperative (ST-SR-IA) and strongly cooperative (ST-MR-IA) tasks using homogenous robots and under task uncertainty. From the literature (Chapter 3), we have observed that there exists many solutions to address ST-SR-IA tasks, few studies address ST-MR-IA tasks, and hardly any address the combination of ST-SR-IA and ST-MR-IA tasks. O. Shehory and S. Kraus (1995) address a similar problem, in which materials of different sizes and weights need to be transported, by proposing a distributed set-partitioning algorithm.

However, in this problem, agents (robots) differ in their capabilities, meaning that the type of operations they can perform differ from agent to agent. In this way, the set of robots is partitioned into subsets, depending on their capabilities. Furthermore, in this algorithm, the average computational complexity is high, considering that each agent computes the coalition values to decide upon the preferred coalitions. In addition, this algorithm scales poorly with the number of agents and tasks to perform. J. Guerrero and G. Oliver (2012) address an ST-MR-IA task assignment problem using an auction-based algorithm in which the robot that discovers first the task becomes the leader and holds an auction to find other robots. As shown, this method drastically reduces the computation complexity. However, in this research, a solution to solve both ST-SR-IA and ST-MR-IA tasks is lacking. Finally, F. Tang and L.E. Parker (2007) present a distributed algorithm called ASyMTRe-D, which can be combined with auction-based algorithms, to enable robots perform both types of tasks, weakly cooperative and strongly cooperative. However, heterogeneous robots are used to perform tasks and coalitions are formed according to the capabilities of robots. Moreover, tasks are assigned sequentially in the experiments. Therefore, as explained by the authors, at time x, task 1 is auctioneed, while at time x+1 and x+2, task 2 and 3 are auctioneed. When the coalitions for these tasks are determined, other tasks are announced. This implies a considerable number of idleness periods for robots. Furthermore, considering that some robots are more capable than others, when these robots are already performing other tasks, the less capable robots need to wait until the accomplishment of said tasks to form coalitions with capable robots.

Therefore, the knowledge gap of this thesis dissertation was to find a solution for robots to perform ST-SR-IA and ST-MR-IA tasks using homogenous robots and reducing periods of idleness and high computations. In order to do so, we have used the method suggested by J. Guerrero and G. Oliver to address ST-MR-IA tasks, adopting a combination of leader-follower and auction-like algorithm. Adding to this work, we have implemented a solution for the dynamic switch of roles for robots to address both ST-SR-IA and ST-MR-IA tasks, within the same application. By using this algorithm, when a robot discovers a ST-SR-IA task, it decides to act in a selfish (non-cooperative) manner, transporting the parcels individually to destination. While, when a robot discovers a ST-MR-IA task, it becomes a leader and starts recruiting followers to operate cooperative transport of parcels. This solution is simple, efficient and involves low communication and computation complexity.

6.2.2 Societal Relevance

From a societal point of view, the developed model is of high interest, since it can be used as a decision-making tool to evaluate different configurations (input parameters) to improve robustness and system effectiveness. This model preserves high degree of generalization, thereby offering the chance to evaluate the multi-robot system in other layouts or changing conditions, such as changing percentages of heavy and high volume payloads. Moreover, the design of different traffic alternatives can be exploited to make trade-offs between system effectiveness, infrastructure usage and robustness. The model can further provide decision-makers with insights into the workspace to use with increasing number of robots and increasing percentages of heavy and high volume parcels. Moreover, it can be also used for comparisons of different technologies (e.g. UAVs and UGVs).

Ultimately, not only postal industries can use the model, but also other logistics operators that require the execution of sorting and indoor/outdoor material transportation operations can exploit its potentialities. For instance, the cooperative transport of goods can be used for the “last mile transportation”. Last mile refers to the last leg of the supply chain, where goods are transported to their final destinations. Recently, famous retail companies have experimented the use of UAVs for delivering goods purchased online to customers. So far, these companies have only researched the use of large aerial vehicles for the transportation of both small and big parcels. However, these companies could take advantage of cooperative and non-cooperative transport behaviors to reduce costs and increase the effectiveness of last mile transport operations. Therefore, some theories and approaches used in this study could be exploited in other fields, such as for the improvement of last mile transport operations and evolvement of effective mini-hubs.

6.3 Discussion: comparing the new multi-robot system with already implemented multi-robot systems

In Chapter 3, we have defined three multi-robot approaches, namely *swarm/collective*, *cooperative* and *networked*. These approaches have been discriminated using ten system dimensions: population size, composition, hardware and software, performance, scalability, flexibility, fault tolerance, reconfigurability (or adaptability) and domains of application. Further, we have associated already implemented multi-robot systems in logistics to these three categories. From this, we have concluded that the Kiva System is an instance of a networked multi-robot system. Robots are equipped with complex hardware components that allow them to have good knowledge of the environment and perform complex transport operations. This system utilizes a multi-agent software control, providing high performance and discrete robustness. However, it does not provide enough flexibility, adaptability and scalability, in view of the dependence on grid-like structures and rigid requirements in terms of max load and shape of objects. The STO Express exhibits very similar characteristics to the Kiva System. This system also provides high performance and satisfying robustness, but lacks in flexibility, scalability and robustness. This system strictly depends on grids and utilizes bar codes placed on the ground for absolute positioning of robots. Moreover, the sorting robots can only handle simple tasks, i.e. transporting small parcels. This system cannot be used with different types of parcels. The Hive from Ocado and AutoStore from Swisslog are other two instances of networked systems, with fully centralized control schemes, high sensing and computational resources. These systems produce high performance, but poor fault tolerance, scalability, flexibility and adaptability.

In comparison to these systems, our multi-robot system eliminates the dependence on grid-like structures providing higher adaptability and scalability. Furthermore, using this system, robots are able to address ST-SR-IA tasks (transportation of small parcels) and ST-MR-IA tasks (transportation of big parcels). This provides the system with higher flexibility. The fault tolerance is increased as deliberative schemes are combined with reactive schemes (hybrid centralized-decentralized control). This provides lower performance, but higher robustness, flexibility, scalability and adaptability. The scalability of the system guarantees profitable system functioning with smaller or larger system scales, as shown in the experimental designs. The flexibility provides the system with the ability to cope with changing percentages of light-low volume / heavy-high volume parcels. System performance can be increased by introducing additional robots or implementing multi-layered sorting centers. The ability of robots to perform concurrently non-cooperative and cooperative tasks enable them to handle any types of task, modularly altering the size and shape of formations. This system marks a milestone towards the design of more flexible, *cooperative* warehouse automation systems.

6.4 Most important limitations

Despite the model provides a satisfactory overview of sorting operations and it preserves some reality, it also brings along important limitations. First, this model is limited to 2D motion and considers robots as point-of-masses, thereby making the simulations physical implementation independent. Therefore, the dynamic physics of robots and the technical features of sensors and actuators are not included in our model. Another limitation of this model is that the energy consumption of robots is not taken into account, as also remarked by an expert in the model validation phase. The energy recharging process of robots can cause serious disruption, especially with high number of robots. In this project, we have decided to not focus on this aspect, which requires advanced scheduling optimization, and we have run simulation for only half hour in which time robots do not need recharging. Another missing feature is the absence of maintenance-related indicators, required by maintenance planners to schedule adequately the maintenance of robots. During the validation phase, an academic expert in logistics also indicated this limitation and underlined its importance at tactical-strategic level. Furthermore, other assumptions made can compromise the real-world effects of the multi-robot systems. For instance, in the implementation of the model, it was assumed that parcels arrive continuously at pick-up buffers, with perfect distribution. In real-sorting operations, certain inbound trucks may arrive earlier than other trucks, meaning that robots can only collect parcels from limited pick-up buffers. Therefore, the demand of pick-up buffers fluctuates within a single shift. This may have an impact on the min-max heuristics algorithm used for the resource allocation of robots to tasks. Another limitation concerns with the exclusion of social aspects from the model. In fact, every time robots arrive at pick-up buffers, someone can place parcels on them. However, in some occasions, robots need to wait longer than expected. Similarly, at drop-off buffers, robots should wait longer time intervals, especially during the replacement of full containers with empty containers. Additionally, the adopted algorithms have not been tested in comparison to other algorithms. Therefore, other algorithms might produce equal or even better results than the one used in our model. Another relevant limitation concerns the technique used for the validation of the model, being expert validation. This technique has significant problems, considering that the opinions of experts are often subjective and biased. When interviewing experts, the answers to questions may deviate from the scope of the project, ascribing unrelated notions from other fields to the model. This problem accentuates even more when consulting experts that were not involved in the model conceptualization phase.

6.5 Future work: recommendations for Prime Vision and possible model extensions

This research should ideally not remain a theoretical work but lead to a system that is factually applicable in real world. Therefore, this work should be followed by prototyping, which serves to test the concepts proposed in this dissertation and provide specifications for a working multi-robot parcel sorting system. After robots are built according to the specifications and with an instinct for cooperative and non-cooperative behaviors, a small-scale pilot can be executed where the actions of a limited number of robots are coordinated in a mixed traffic configuration. Experimental designs should be then executed, with altering number of robots and percentages of light/heavy and low/high volume parcels, to evaluate the results and compare them with those obtained in our project. The number of robots in formations should range from 2 to 4 according to the different types of parcels to handle. As explained in this thesis, the developed algorithm scales suitably, thus allowing for the creation of different formations. If the small-scale pilot is successful, a large-scale pilot can be performed, with higher number of robots and higher differences in types of parcels to address. In the second pilot, other traffic configurations could be tested to evaluate the advantages and disadvantages of each design alternative and find an optimal configuration. According to the results of this thesis, mixed traffic configuration could be used with a number of robots inferior than 150. Over that number of robots, a Two-Layered or Highway should be implemented. Regarding the type of robots to use, it is appropriate to start with UGVs that guarantee higher safety and lower energy consumption. In a later stage, the use of UAVs should be tested, once adequate solutions to defeat the energy problem and to increase the payload capacity are found. An indoor application of UAVs has an enormous potential, given that a similar system could be implemented in smaller workspaces, with shorter distances between pick-up and drop-off buffers that would also guarantee higher system effectiveness. Furthermore, UAVs can execute the loading and unloading operations of parcels without human or other robotic assistance. With the adoption of UAVs, the Two-Layered traffic configuration should be used in order to increase the predictability and safety of the system.

As already argued, the model is capable of providing parcel operators with a large number of information. However, as indicated in the limitations paragraph, further improvements can be made. The energy usage of robots should be added to the model, and an optimized scheduling for the recharging process of robots should be developed. This factor is also valuable for the application of the auction-based algorithm to build formations of robots with high level of energy.

In addition, the distribution of parcels across pick-up buffers should not be uniform. The implementation of a non-uniform parcel distribution will have an effect on the min-max heuristic algorithm used for the allocation of robots to pick-up buffers. Using this algorithm, robots decide to drive towards pick-up buffers regardless on the amount of parcels a buffer contains. For the enhancement of the min-max heuristic, we propose the assignment of different weights to pick-up buffers based on the amount of parcels contained and give robots a certain probability to move towards buffers with higher weights. The algorithm should retain some randomness to prevent robots to choose the same buffers at the same time, leading to overcrowding stations. Therefore, the algorithm should take into consideration these three key factors: randomness; non-uniform distribution of parcels; min-max waiting time to prompt more robots to move towards heavy and high volume parcels. However, this solution needs to be tested by means of simulation to evaluate its adequacy. In the event certain pick-up buffers do not contain any parcel, robots should consider these buffers as not working and exclude them from their allocation decision. This strategy was implemented in the experimental design 5, which considers disruptive scenarios with damaged pick-up buffers.

A larger variety of parcels should be introduced, such as light / high volume, heavy / low volume parcels or of medium size and weight. In this way, a different subset of robots should cooperate to transport different types of parcels. The formation of robots might be also altered according to the shape of parcels. Additionally, the layout of the sorting center can be changed to test the effectiveness and flexibility of robots in diverse conditions. Besides, the position of pick-up and drop-off buffers could be modified to find an optimal configuration that shortens the distances travelled of robots and reduces the level of congestion. Furthermore, human factors should also be included in the model to increase its reality. As described in the limitations, robots should be able to wait longer in certain circumstances without bringing about disruptive situations. Another potential improvement concerns the developed assistance mechanism, which could be improved with the usage of homogenous robots in place of heterogeneous robots. In this way, the multi-robot parcel sorting system can achieve full autonomy. These model extensions can pave the way towards continuous improvement. Other future works could involve the maintenance of robots and the economic feasibility of this system. Both aspects were not considered in this research. Furthermore, researchers can use this research as an instance towards additional experiments using similar or different designs, providing further insights into the functionality of this system.

References

Amditis, A., Thomaidis, G., Maroudis, P., Lytrivis, P., & Karaseitanidis, G. (2012). Multiple hypothesis tracking implementation. In *Laser Scanner Technology*. InTech.

Angel, B. W. F., Van Damme, D. A., Ivanovskaia, A., Lenders, R. J. M., & Veldhuijzen, R. S. (2006). Warehousing space in Europe: meeting tomorrow's demand. *Prologis/Cap Gemini, Utrecht*.

Arai, T., Pagello, E., & Parker, L. E. (2002). Advances in multi-robot systems. *IEEE Transactions on robotics and automation*, *18*(5), 655-661.

Arkin, R. C. (1998). *Behavior-based robotics*. MIT press.

Barca, J. C., & Sekercioglu, Y. A. (2013). Swarm robotics reviewed. *Robotica*, *31*(3), 345-359.

Bayındır, L. (2016). A review of swarm robotics tasks. *Neurocomputing*, *172*, 292-321.

Beni, G., & Wang, J. (1993). Swarm intelligence in cellular robotic systems. In *Robots and Biological Systems: Towards a New Bionics?* (pp. 703-712). Springer, Berlin, Heidelberg.

Bhattacharya, S., Likhachev, M., & Kumar, V. (2010). *Distributed path consensus algorithm*. Technical Report MS-CIS-10-07, University of Pennsylvania.

Bloss, R. (2013). Automation pushes the envelope of postal mail handling efficiency. *Assembly Automation*, *33*(1), 3-7.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems* (No. 1). Oxford university press.

Boysen, N., Fliedner, M., & Scholl, A. (2010). Scheduling inbound and outbound trucks at cross docking terminals. *OR spectrum*, *32*(1), 135-161.

Bozer, Y. A., & Carlo, H. J. (2008). Optimizing inbound and outbound door assignments in less-than-truckload crossdocks. *IIE Transactions*, *40*(11), 1007-1018.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, *7*(1), 1-41.

Breitenmoser, A., Schwager, M., Metzger, J. C., Siegwart, R., & Rus, D. (2010, May). Voronoi coverage of non-convex environments with a group of networked robots. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (pp. 4982-4989). IEEE.

Browne, J., Dubois, D., Rathmill, K., Sethi, S. P., & Stecke, K. E. (1984). Classification of flexible manufacturing systems. *The FMS magazine*, *2*(2), 114-117.

Brutschy, A., Pini, G., Pinciroli, C., Birattari, M., & Dorigo, M. (2014). Self-organized task allocation to sequentially interdependent tasks in swarm robotics. *Autonomous agents and multi-agent systems*, *28*(1), 101-125.

Cai, A. H., Fukuda, T., Arai, F., Ueyama, T., & Sakai, A. (1995, May). Hierarchical control architecture for cellular robotic system-simulations and experiments. In *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on* (Vol. 1, pp. 1191-1196). IEEE.

Cai, C., Yang, C., Zhu, Q., & Liang, Y. (2007, August). Collision avoidance in multi-robot systems. In *Mechatronics and Automation, 2007. ICMA 2007. International Conference on* (pp. 2795-2800). IEEE.

Cali, F., Conti, M., & Gregori, E. (2000). IEEE 802.11 protocol: design and performance evaluation of an adaptive backoff mechanism. *IEEE journal on selected areas in communications*, *18*(9), 1774-1786.

Chandy, K. M., Misra, J., & Haas, L. M. (1983). Distributed deadlock detection. *ACM Transactions on Computer Systems (TOCS)*, *1*(2), 144-156.

Chen, Y., & Kobayashi, H. (2002). Signal strength based indoor geolocation. In *Communications, 2002. ICC 2002. IEEE International Conference on* (Vol. 1, pp. 436-439). IEEE.

Consolini, L., Morbidi, F., Prattichizzo, D., & Tosques, M. (2008). Leader–follower formation control of nonholonomic mobile robots with input constraints. *Automatica*, *44*(5), 1343-1349.

Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Transactions on robotics and Automation*, *20*(2), 243-255.

Crew, M., Parcu, P. L., & Brennan, T. (Eds.). (2017). *The Changing Postal and Delivery Sector: Towards A Renaissance*. Springer.

D'Andrea, R. (2012). Guest editorial: A revolution in the warehouse: A retrospective on kiva systems and the grand challenges ahead. *IEEE Transactions on Automation Science and Engineering*, *9*(4), 638-639.

D'Souza, K. A., & Khator, S. K. (1994). A survey of Petri net applications in modeling controls for automated manufacturing systems. *Computers in industry*, *24*(1), 5-16.

Davidsson, P. (2000). Multi agent based simulation: beyond social simulation. In *Multi-Agent-*

Based Simulation (pp. 97-107). Springer Berlin Heidelberg.

Desai, J. P., Ostrowski, J., & Kumar, V. (1998, May). Controlling formations of multiple mobile robots. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on* (Vol. 4, pp. 2864-2869). IEEE.

Desai, J. P., Ostrowski, J. P., & Kumar, V. (2001). Modeling and control of formations of nonholonomic mobile robots. *IEEE transactions on Robotics and Automation*, *17*(6), 905-908.

Discant, A., Rogozan, A., Rusu, C., & Bensrhair, A. (2007, May). Sensors for obstacle detection-a survey. In *Electronics Technology, 30th International Spring Seminar on* (pp. 100-105). IEEE.

Dorigo, M., Birattari, M., & Brambilla, M. (2014). Swarm robotics. *Scholarpedia*, *9*(1), 1463.

Dym, C. L., Little, P., Orwin, E. J., & Spjut, E. (2009). *Engineering design: A project-based introduction*. John Wiley and sons.

Duchoň, F., Babinec, A., Kajan, M., Beňo, P., Florek, M., Fico, T., & Jurišica, L. (2014). Path planning with modified a star algorithm for a mobile robot. *Procedia Engineering*, *96*, 59-69.

Dudek, G., Jenkin, M. R., Milios, E., & Wilkes, D. (1996). A taxonomy for multi-agent robotics. *Autonomous Robots*, *3*(4), 375-397.

Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: part I. *IEEE robotics & automation magazine*, *13*(2), 99-110.

Elango, M., Nachiappan, S., & Tiwari, M. K. (2011). Balancing task allocation in multi-robot systems using K-means clustering and auction based mechanisms. *Expert Systems with Applications*, *38*(6), 6486-6491.

El Zaher, M., Contet, J. M., Gechter, F., & Koukam, A. (2012). Echelon platoon organisation: A distributed approach based on 2-spring virtual links. *Artificial Intelligence: Methodology, Systems, and Applications*, 246-255.

Farinelli, A., Iocchi, L., & Nardi, D. (2004). Multirobot systems: a classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *34*(5), 2015-2028.

Farinelli, A., Zanotto, E., & Pagello, E. (2017). Advanced approaches for multi-robot coordination in logistic scenarios. *Robotics and Autonomous Systems*, *90*, 34-44.

Fedtke, S., & Boysen, N. (2014). Layout planning of sortation conveyors in parcel distribution centers. *Transportation Science*, *51*(1), 3-18.

Fierro, R., Das, A., Spletzer, J., Esposito, J., Kumar, V., Ostrowski, J. P., ... & Lee, I. (2002). A framework and architecture for multi-robot coordination. *The International Journal of Robotics Research*, *21*(10-11), 977-995.

Fikse, K., Haneyah, S. W. A., & Schutten, J. M. J. (2012). *Improving the performance of sorter systems by scheduling inbound containers*. Beta Working Paper series 376, 2012, http://cms. ieis. tue. nl/Beta/Files/WorkingPapers/wp_376. pdf.

Firby, R. J. (1990). *Adaptive execution in complex dynamic worlds* (Doctoral dissertation, Yale University).

Fox, D., Ko, J., Konolige, K., Limketkai, B., Schulz, D., & Stewart, B. (2006). Distributed multirobot exploration and mapping. *Proceedings of the IEEE*, *94*(7), 1325-1339.

Gale, D. (1960). *The theory of linear economic models*. University of Chicago press.

Garcia-Romeu-Martinez, M. A., Singh, S. P., Cloquell-Ballester, V. A., & Saha, K. (2007). Measurement and analysis of international air parcel shipping environment for DHL and FedEx between Europe and United States. *Packaging Technology and Science*, *20*(6), 421-429.

Gerkey, B. P., & Matarić, M. J. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, *23*(9), 939-954.

Ghommam, J., Mehrjerdi, H., Saad, M., & Mnif, F. (2010). Formation path following control of unicycle-type mobile robots. *Robotics and Autonomous Systems*, *58*(5), 727-736.

Grassé, P. P. (1959). La reconstruction du nid et les coordinations interindividuelles chezBellicositermes natalensis etCubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes sociaux*, *6*(1), 41-80.

Guerrero, J., & Oliver, G. (2012). Multi-robot coalition formation in real-time scenarios. *Robotics and Autonomous Systems*, *60*(10), 1295-1307.

Guizzo, E. (2008). Three engineers, hundreds of robots, one warehouse. *IEEE spectrum*, *45*(7), 26-34.

Hamberg, R., & Verriet, J. (2012). The Falcon Project: Model-Based Design of Automated Warehouses. In *Automation in Warehouse Development* (pp. 3-13). Springer London.

Hazard, C. J., Wurman, P. R., & D'Andrea, R. (2006, July). Alphabet soup: A testbed for studying resource allocation in multi-vehicle systems. In *Proceedings of the 2006 AAAI Workshop on Auction Mechanisms for Robot Coordination* (pp. 23-30).

Howard, A. (2006). Multi-robot simultaneous localization and mapping using particle filters. *The International Journal of Robotics Research*, *25*(12), 1243-1256.

Hungerford, K., Dasgupta, P., & Guruprasad, K. R. (2016). A repartitioning algorithm to guarantee complete, non-overlapping planar coverage with multiple robots. In *Distributed Autonomous Robotic Systems* (pp. 33-48). Springer, Tokyo.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, *2*(1-2), 83-97.

Hichri, B., Adouane, L., Fauroux, J. C., Mezouar, Y., & Doroftei, I. (2016). Cooperative mobile robot control architecture for lifting and transportation of any shape payload. In *Distributed Autonomous Robotic Systems* (pp. 177-191). Springer, Tokyo.

Hintenaus, P. (2015). Short-Range Radar. In *Engineering Embedded Systems* (pp. 265-284). Springer International Publishing.

Huntsberger, T., Pirjanian, P., Trebi-Ollennu, A., Nayar, H. D., Aghazarian, H., Ganino, A. J., ... & Schenker, P. S. (2003). CAMPOUT: A control architecture for tightly coupled coordination of multirobot systems for planetary surface exploration. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *33*(5), 550-559.

Iocchi, L., Nardi, D., & Salerno, M. (2000, August). Reactivity and deliberation: a survey on multi-robot systems. In *Workshop on Balancing Reactivity and Social Deliberation in Multi-Agent Systems* (pp. 9-32). Springer, Berlin, Heidelberg.

Ivanov, P., & Shell, D. A. (2017). Hark! Who goes there? Concurrent association of communication channels for multiple mobile robots. *Autonomous Robots*, *41*(4), 811-829.

Jadbabaie, A., Lin, J., & Morse, A. S. (2003). Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on automatic control*, *48*(6), 988-1001.

Jager, M., & Nebel, B. (2001). Decentralized collision avoidance, deadlock detection, and deadlock resolution for multiple mobile robots. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on* (Vol. 3, pp. 1213-1219). IEEE.

Johnson, M. E., & Meller, R. D. (2002). Performance analysis of split-case sorting systems. *Manufacturing & Service Operations Management*, *4*(4), 258-274.

Julian, B. J. (2009). *An embedded controller for quad-rotor flying robots running distributed algorithms* (Doctoral dissertation, Massachusetts Institute of Technology).

Kamel, M. A., & Zhang, Y. (2015, May). Decentralized leader-follower formation control with obstacle avoidance of multiple unicycle mobile robots. In *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on* (pp. 406-411). IEEE.

Kernbach, S. (Ed.). (2013). *Handbook of collective robotics: fundamentals and challenges*. CRC Press.

Khamis, A., Hussein, A., & Elmogy, A. (2015). Multi-robot task allocation: A review of the state-of-the-art. In *Cooperative Robots and Sensor Networks 2015* (pp. 31-51). Springer International Publishing.

Kheir, N. (Ed.). (1995). *Systems modeling and computer simulation* (Vol. 94). CRC Press.

Kloetzer, M., Mahulea, C., & Colom, J. M. (2013, September). Petri net approach for deadlock prevention in robot planning. In *Emerging Technologies & Factory Automation (ETFA), 2013 IEEE 18th Conference on*(pp. 1-4). IEEE.

Klutke, G. A., Kiessler, P. C., & Wortman, M. A. (2003). A critical look at the bathtub curve. *IEEE Transactions on Reliability*, *52*(1), 125-129.

Koch, P., May, S., Schmidpeter, M., Kühn, M., Pfitzner, C., Merkl, C., ... & Nüchter, A. (2016). Multi-robot localization and mapping based on signed distance functions. *Journal of Intelligent & Robotic Systems*, *83*(3-4), 409-428.

Kong, C. S., Peng, N. A., & Rekleitis, I. (2006, May). Distributed coverage with multi-robot system. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on* (pp. 2423-2429). IEEE.

Kube, C. R., & Bonabeau, E. (2000). Cooperative transport by ants and robots. *Robotics and autonomous systems*, *30*(1), 85-101.

Kube, C. R., & Zhang, H. (1993). Collective robotics: From social insects to robots. *Adaptive behavior*, *2*(2), 189-218.

Lattarulo, V., & Parks, G. T. (2012, June). A preliminary study of a new multi-objective optimization algorithm. In *Evolutionary Computation (CEC), 2012 IEEE Congress on* (pp. 1-8). IEEE.

Lee, B. H., & Lee, C. G. (1987). Collision-free motion planning of two robots. *IEEE Transactions on Systems, Man, and Cybernetics*, *17*(1), 21-32.

Leonard, N. E., & Fiorelli, E. (2001). Virtual leaders, artificial potentials and coordinated control of groups. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on* (Vol. 3, pp. 2968-2973). IEEE.

Lin, L., & Zheng, Z. (2005, April). Combinatorial bids based multi-robot task allocation method. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on* (pp. 1145-1150). IEEE.

Liu, W., Winfield, A. F., Sa, J., Chen, J., & Dou, L. (2007). Towards energy optimization: Emergent task allocation in a swarm of foraging robots. *Adaptive behavior*, *15*(3), 289-305.

Luck, M., McBurney, P., Shehory, O., & Willmott, S. (2005). *Agent technology: computing as interaction (a roadmap for agent based computing)*. University of Southampton.

Luo, L., Chakraborty, N., & Sycara, K. (2015). Provably-good distributed algorithm for constrained multi-robot task assignment for grouped tasks. *IEEE Transactions on Robotics*, *31*(1), 19-30.

MAINO, C. (2014). Sistemi automatizzati innovativi per le attività di stoccaggio e prelievo all'interno dei centri distributivi.

Mas, I., & Kitts, C. (2010, July). Centralized and decentralized multi-robot control methods using the cluster space control framework. In *Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on*(pp. 115-122). IEEE.

Marino, A., Parker, L. E., Antonelli, G., & Caccavale, F. (2013). A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling. *Journal of Intelligent & Robotic Systems*, *71*(3-4), 423-444.

Martínez, D. L., & Halme, A. (2016). MarSim, a simulation of the MarsuBots fleet using NetLogo. In *Distributed Autonomous Robotic Systems* (pp. 79-87). Springer Japan.

McAree, P., Bodin, L., Ball, M., & Segars, J. (2006). Design of the federal express large package sort facility. *Annals of Operations Research*, *144*(1), 133-152.

McWilliams, D. L., Stanfield, P. M., & Geiger, C. D. (2005). The parcel hub scheduling problem: A simulation-based solution approach. *Computers & Industrial Engineering*, *49*(3), 393-412.

Navarro, I., & Matía, F. (2012). An introduction to swarm robotics. *ISRN Robotics*, *2013*.

Neuman, T. R. (1985). Intersection channelization design guide. *NCHRP Report*, (279).

Niazi, M., & Hussain, A. (2011). Agent-based computing from multi-agent systems to agent-based models: a visual survey. *Scientometrics*, *89*(2), 479.

Noguchi, N., Will, J., Reid, J., & Zhang, Q. (2004). Development of a master–slave robot system for farm operations. *Computers and Electronics in agriculture*, *44*(1), 1-19.

Olfati-Saber, R., & Murray, R. M. (2002, December). Distributed structural stabilization and tracking for formations of dynamic multi-agents. In *Decision and Control, 2002,* *Proceedings of the 41st IEEE Conference on* (Vol. 1, pp. 209-215). IEEE.

Oyekan, J., & Huosheng, H. (2009, August). Toward bacterial swarm for environmental monitoring. In *Automation and Logistics, 2009. ICAL'09. IEEE International Conference on* (pp. 399-404). IEEE.

Parker, L. E. (1995). On the design of behavior-based multi-robot teams. *Advanced Robotics*, *10*(6), 547-578.

Parker, L. E. (1998). ALLIANCE: An architecture for fault tolerant multirobot cooperation. *IEEE transactions on robotics and automation*, *14*(2), 220-240.

Parker, L. E., & Tang, F. (2006). Building multirobot coalitions through automated task solution synthesis. *Proceedings of the IEEE*, *94*(7), 1289-1305.

Parker, L. E. (2008). Distributed intelligence: Overview of the field and its application in multi-robot systems. *Journal of Physical Agents*, *2*(1), 5-14.

Parker, L. E. (2008). Multiple mobile robot systems. In *Springer Handbook of Robotics* (pp. 921-941). Springer Berlin Heidelberg.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, *24*(3), 45-77.

Portugal, D., & Rocha, R. P. (2013). Distributed multi-robot patrol: A scalable and fault-tolerant framework. *Robotics and Autonomous Systems*, *61*(12), 1572-1587.

Qu, Z. (2009). *Cooperative control of dynamical systems: applications to autonomous vehicles*. Springer Science & Business Media.

Ragsdale, C. T. (2011). *Managerial decision modeling*.

Rasshofer, R. H., & Gresser, K. (2005). Automotive radar and lidar systems for next generation driver assistance functions. *Advances in Radio Science*, *3*(B. 4), 205-209.

Regele, R., & Levi, P. (2006, October). Cooperative multi-robot path planning by heuristic priority adjustment. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on* (pp. 5954-5959). IEEE.

Robertson, S. (2001). Requirements trawling: techniques for discovering requirements. *International Journal of Human-Computer Studies*, *55*(4), 405-421.

Sadler, J. (2006). *The future of European food and drinks retailing: Implications of fast growth formats and private label for brand manufacturers*. Business Insights Limited.

Sanchez, J. G., Pascual, M. M., Marinero, J. F., Rojo, F. G., Turiel, J. P., & Gonzlez, F. G. (2002). Analysis of utilization and throughput in a multirobot system. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on* (Vol. 1, pp. 96-101). IEEE.

Şahin, E. (2004, July). Swarm robotics: From sources of inspiration to domains of application. In *International workshop on swarm robotics* (pp. 10-20). Springer, Berlin, Heidelberg.

Sahota, M. K. (1994, October). Reactive deliberation: An architecture for real-time intelligent control in dynamic environments. In *AAAI* (pp. 1303-1308).

Sargent, R. G. (2005, December). Verification and validation of simulation models. In *Proceedings of the 37th conference on Winter simulation* (pp. 130-143). winter simulation conference.

Schwager, M., Rus, D., & Slotine, J. J. (2009). Decentralized, adaptive coverage control for networked robots. *The International Journal of Robotics Research*, *28*(3), 357-375.

Schreckenghost, D., Bonasso, P., Kortenkamp, D., & Ryan, D. (1998, May). Three tier architecture for controlling space life support systems. In *Intelligence and Systems, 1998. Proceedings., IEEE International Joint Symposia on* (pp. 195-201). IEEE.

Shehory, O., & Kraus, S. (1995, August). Task allocation via coalition formation among autonomous agents. In *IJCAI (1)* (pp. 655-661).

Shiomi, M., Kamei, K., Kondo, T., Miyashita, T., & Hagita, N. (2013, November). Robotic service coordination for elderly people and caregivers with ubiquitous network robot platform. In *Advanced Robotics and its Social Impacts (ARSO), 2013 IEEE Workshop on* (pp. 57-62). IEEE.

Shirase, K., & Aoyagi, S. (Eds.). (2009). *Service Robotics and Mechatronics: Selected Papers of the International Conference on Machine Automation ICMA2008*. Springer Science & Business Media.

Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). *Introduction to autonomous mobile robots*. MIT press.

Silberschatz, A., Galvin, P. B., Gagne, G., & Silberschatz, A. (1998). *Operating system concepts* (Vol. 4). Reading: Addison-wesley.

Simmons, R., Apfelbaum, D., Burgard, W., Fox, D., Moors, M., Thrun, S., & Younes, H. (2000, July). Coordination for multi-robot exploration and mapping. In *AAAI/IAAI* (pp. 852-858).

Sira-Ramírez, H., & Castro-Linares, R. (2010, December). Trajectory tracking for non-holonomic cars: A linear approach to controlled leader-follower formation. In *Decision and Control (CDC), 2010 49th IEEE Conference on*(pp. 546-551). IEEE.

Specht, E. (2010). The best known packings of equal circles in a square.

Spletzer, J., Das, A. K., Fierro, R., Taylor, C. J., Kumar, V., & Ostrowski, J. P. (2001). Cooperative localization and control for multi-robot manipulation. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on* (Vol. 2, pp. 631-636). IEEE.

Sun, D., Kleiner, A., & Nebel, B. (2014, May). Behavior-based multi-robot collision avoidance. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on* (pp. 1668-1673). IEEE.

Sung, C., Ayanian, N., & Rus, D. (2013, May). Improving the performance of multi-robot systems by task switching. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on* (pp. 2999-3006). IEEE.

Tanenbaum, A. S. (2003). Computer networks, 4-th edition. *ed: Prentice Hall*.

Tang, F., & Parker, L. E. (2007, April). A complete methodology for generating multi-robot task solutions using asymtre-d and market-based task allocation. In *Robotics and Automation, 2007 IEEE International Conference on* (pp. 3351-3358). IEEE.

Tarău, A. N., De Schutter, B., & Hellendoorn, J. (2009). Model-based control for throughput optimization of automated flats sorting machines. *Control Engineering Practice*, *17*(6), 733-739.

Thrun, S. (2002). Robotic mapping: A survey, Exploring Artificial Intelligence in the New Millenium, Chapter I.

Van Dam, K. H., Nikolic, I., & Lukszo, Z. (Eds.). (2012). *Agent-based modelling of socio-technical systems* (Vol. 9). Springer Science & Business Media.

Verret, S. (2005). Current state of the art in multirobot systems. *Defence Research and Development Canada-Suffield*, 3.

Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I., & Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical review letters*, *75*(6), 1226.

Vig, L., & Adams, J. A. (2006). Multi-robot coalition formation. *IEEE transactions on robotics*, *22*(4), 637-649.

Volpe, R., Nesnas, I., Estlin, T., Mutz, D., Petras, R., & Das, H. (2001). The CLARAty architecture for robotic autonomy. In *Aerospace Conference, 2001, IEEE Proceedings.* (Vol. 1, pp. 1-121). IEEE.

Wagner, G., & Choset, H. (2011, September). M*: A complete multirobot path planning algorithm with performance bounds. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on* (pp. 3260-3267). IEEE.

Wang, Z., & Schwager, M. (2016). Multi-robot manipulation without communication. In *Distributed Autonomous Robotic Systems* (pp. 135-149). Springer, Tokyo.

Wang, X., Kloetzer, M., Mahulea, C., & Silva, M. (2015, December). Collision avoidance of mobile robots by using initial time delays. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on* (pp. 324-329). IEEE.

Wen, M. (2004). E-commerce, productivity, and fluctuation. *Journal of Economic Behavior & Organization*, *55*(2), 187-206.

Werners, B., & Wülfing, T. (2010). Robust optimization of internal transports at a parcel sorting center operated by Deutsche Post World Net. *European Journal of Operational Research*, *201*(2), 419-426.

Wooldridge, M. (1998). Agent-based computing. *Interoperable Communication Networks*, *1*, 71-98.

Wurman, P. R., D'Andrea, R., & Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI magazine*, *29*(1), 9.

Yan, Z., Jouandeau, N., & Cherif, A. A. (2013). A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, *10*(12), 399.

Yang, Z., Kobayashi, T., & Katayama, T. (2000). *Development of an intersection collision warning system using DGPS* (No. 2000-01-1301). SAE Technical Paper.

Yokoo, M., & Hirayama, K. (2000). Algorithms for distributed constraint satisfaction: A review. *Autonomous Agents and Multi-Agent Systems*, *3*(2), 185-207.

Yunardi, R. T. (2015, October). Contour-based object detection in Automatic Sorting System for a parcel boxes. In *Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA), 2015 International Conference on* (pp. 38-41). IEEE.

Zhang, C., Noguchi, N., & Yang, L. (2016). Leader–follower system using two robot tractors to improve work efficiency. *Computers and Electronics in Agriculture*, *121*, 269-281.

Zhang, Y., & Mehrjerdi, H. (2013, May). A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations. In *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on* (pp. 1087-1096). IEEE.

Zhou, Y., Hu, H., Liu, Y., & Ding, Z. (2017). Collision and Deadlock Avoidance in Multirobot Systems: A Distributed Approach. *IEEE Transactions on Systems, Man, and*

Cybernetics: Systems.

Zlot, R. M., & Stentz, A. (2006). An auction-based approach to complex task allocation for multirobot teams.

Zufferey, J. C., Hauert, S., Stirling, T., Leven, S., Roberts, J., & Floreano, D. (2013). *Aerial collective systems* (No. EPFL-CHAPTER-153134, pp. 609-660). Pan Stanford.

Appendix A

A.1 Pseudo-code distributed mapping algorithm

```
;; Collective mapping

;;initial setup
Ask world space: "divide area in equal squared-shape areas"
calculate total number of robots / squared-shape areas
Ask total number of robots / squared-shape parts: "distribute among squared-shape areas"
Ask robots in each area: "create a circular formation"

start mapping
set "mapping completed" = false
foreach robot in each area
        set speed = 1.5 m/s
        set pick up list = [empty]
        set drop off list = [empty]
        move forward
        if "any pick up buffer in sensor sight"
                do while "pick up list length < pick up buffers"
                        if "pick up buffer not in the pick up list"
                                save pick up buffer in pick up list
                end while
                ask robots in same area:
                        if "pick up buffer not in the pick up list"
                                save pick up buffer in pick up list
        if "any drop off buffer in sensor sight"
                do while "drop off list length < drop off buffers"
                        if "drop off buffer not in the pick up list"
                                save pick up buffer in pick up list
                end while
                ask robots in same area:
                        if "drop off buffer not in the pick up list"
                                save drop off buffer in pick up list
        if "all squared-shape area explored"
                ask "all other robots not in my area"
                        if "pick up buffers in my list not in your pick up list"
                                save pick up buffers in your pick up list
                        if "drop off buffers in my list not in your drop off list"
                                save drop off buffers in your drop off list
        if "all pick up buffers in pick up list AND all drop off buffers in drop off list"
                set "mapping completed" = true
end mapping
```

A.2 Flowchart distributed mapping algorithm

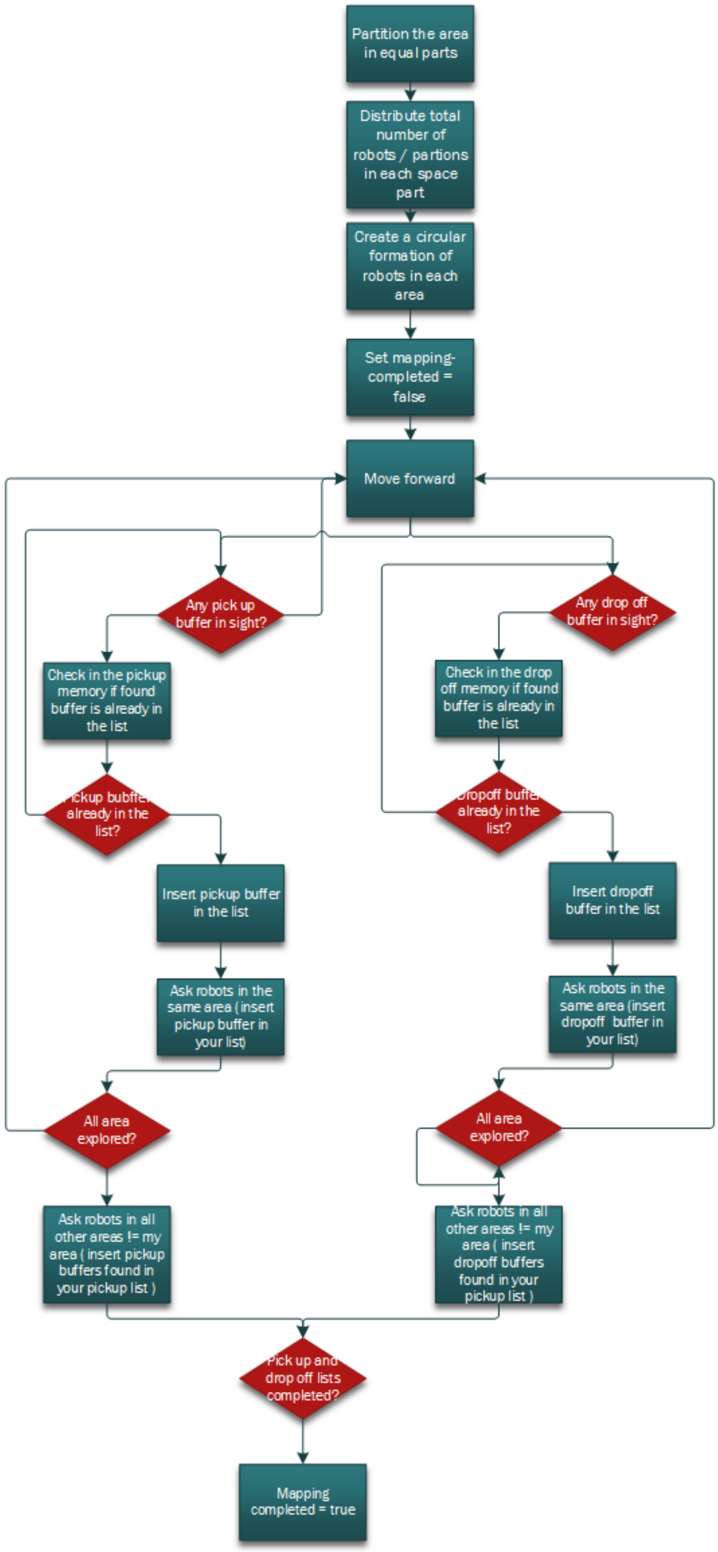


Figure 50: Flowchart distributed mapping algorithm

Appendix B

B.1 Pseudo-code queueing algorithm

```
;; Queueing Management

start queuing

set "semaphore1" = 0
set "semaphore2" = 0

if "semaphore1 = 0 or semaphore2 = 0"
        set "in queue" = false
        set "temporary" = 0
        if "in queue" = false
                do while "in queue" = false AND "temporary" < length of queue
                        if "not other robots on position of queue = temporary
                                move to position of queue = temporary
                                set"in queue" = true
                        else
                                set "temporary" = temporary + 1
                end while
        else
                if "not other robots on position ahead AND position of agent != first position in queue"
                        move to position ahead
                else
                        get "weight" of parcel on pickup buffer
                        if "weight of parcel > payload capacity"
                                get "number of robots" in pickup buffer
                                "minimum number of robots" = integer of "weight of parcel / payload capacity of a robot"
                                if "number of robots < minimum number of robots"
                                        set "semaphore1" = 1
                                else
                                        set "semaphore1" = 0
                        else
                        set "semaphore2" = 1
else
        move to pickup buffer

end queuing
```

B.2 Flowchart queueing algorithm

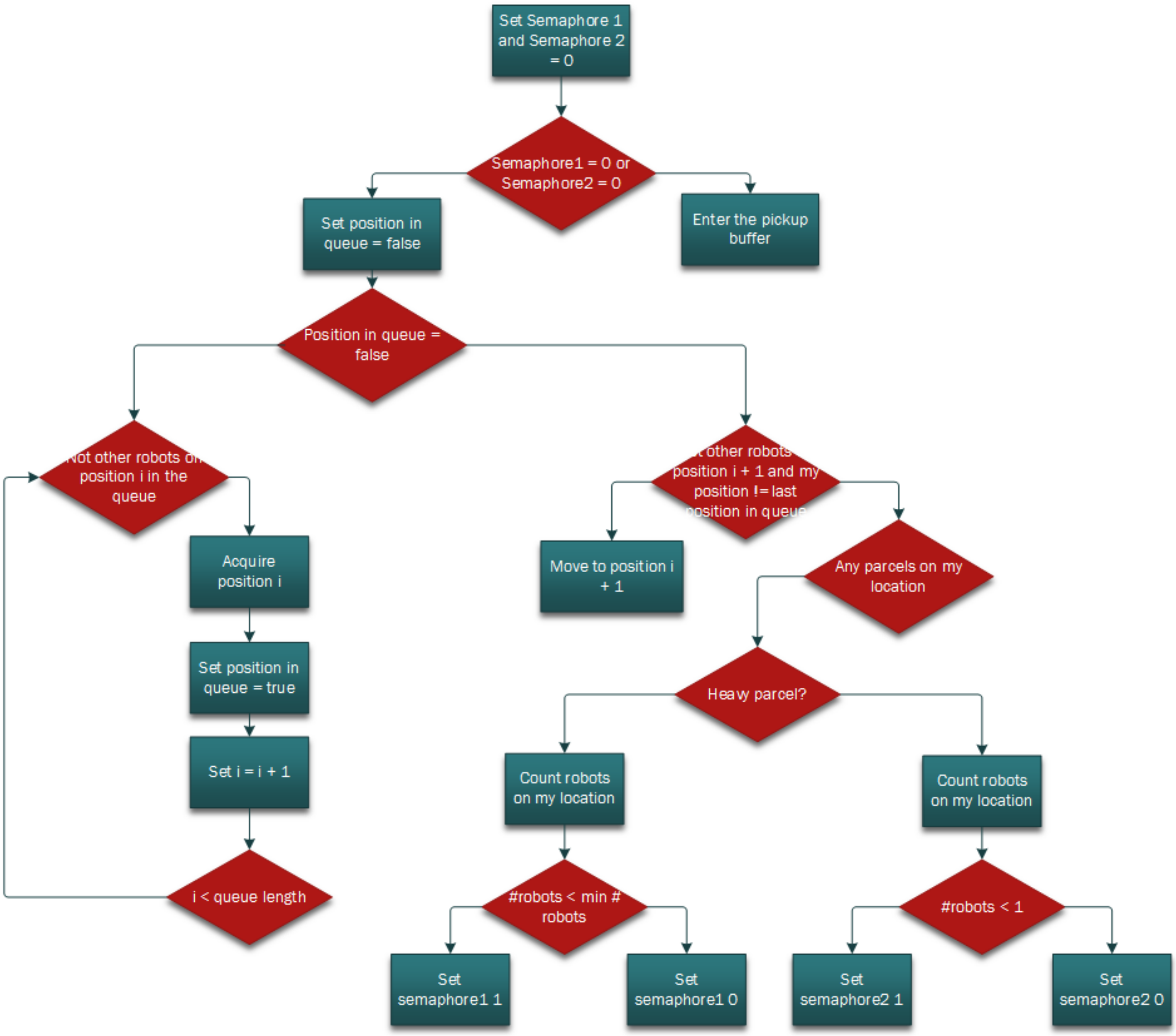


Figure 51: Flowchart queuing algorithm

Appendix C

C.1 Pseudo-code collision avoidance algorithm

```
;; Collision-avoidance in the highway design choice

calculate shortest path to drop off buffer
set speed = 1.5 m/s
move forward towards drop off buffer
set "priority" = 1

start collision-avoidance

if "other robots within sensor sight with priority = 1"
        set speed = 0 m/s
        get "steering angle" of other robot
        if "side collision"
                move backwards
                wait for other robot to pass
        else ;;frontal collision
                if "other robot is going to the right"
                        move to the left
                else
                        move to the right
else
        set speed = 1.5 m/s
        move forward towards drop-off buffer
        if "drop off buffer reached"
                set "loadStatus" = unloaded
                set "priority" = 0
                move towards first reference point outside transport field
                if "other robots with priority = 0 in sensor sight"
                        set speed = 0 m/s
                        wait until robot moves ahead
                else
                        set speed = 1.5 m/s
                        move towards next reference point
                        if "last reference point reached"
                                move towards pick up buffer
                                if "other robot with priority = 1 in sensor sight"
                                        set speed = 0 m/s
                                        wait until robot with higher priority passes
                                else
                                        set speed = 1.5 m/s
                                        if "queue of pick up buffer reached"
                                                enter queue

end collision avoidance

;; Collision-avoidance in the mixed-traffic design choice

calculate shortest path to drop off buffer
set speed = 1.5 m/s
move forward towards drop off buffer

start collision avoidance
```

```
if "other robots within sensor sight"
        set speed = 0 m/s
        get "steering angle" of other robot
        if "side collision"
                move backwards
                wait for other robot to pass
        else ;;frontal collision
                if "other robot is going to the right"
                        move to the left
                else
                        move to the right
else
        set speed = 1.5 m/s
        move forward towards drop-off buffer
        if "drop off buffer reached"
                set "loadStatus" = unloaded
                set "my pickup buffer" = one of pick up buffers in memory list
                compute shortest path to pick up buffer
                move forward towards pick up buffer
                if "other robots in sensor sight"
                        set speed = 0 m/s
                        wait until robot moves ahead
                else
                        set speed = 1.5 m/s
                        move forward towards pick up buffer
                        if "other robots within sensor sight"
                                set speed = 0 m/s
                                get "steering angle" of other robot
                                if "side collision"
                                        move backwards
                                        wait for other robot to pass
                                else ;;frontal collision
                                        if "other robot is going to the right"
                                                move to the left
                                        else
                                                move to the right
                        else
                                set speed = 1.5 m/s
                                move forward toward pick up buffer
                                if "queue of my pickup buffer reached"
                                        enter queue of my pickup buffer

end collision avoidance

;; Collision-avoidance in the 2-floor design choice

calculate shortest path to drop off buffer
set speed = 1.5 m/s
move forward towards drop off buffer
set "priority" = 1

start collision avoidance

if "other robots within sensor sight with priority = 1"
        set speed = 0 m/s
        get "steering angle" of other robot
        if "side collision"
                move backwards
                wait for other robot to pass
```

```
        else ;;frontal collision
                if "other robot is going to the right"
                        move to the left
                else
                        move to the right
else
        set speed = 1.5 m/s
        move forward towards drop-off buffer
        if "drop off buffer reached"
                set "loadStatus" = unloaded
                set "priority" = 0
                move on the ramp
                move forward on the ramp
                if "second floor reached"
                        set "my pickup buffer" = one of pick up buffers in memory list
                        compute shortest path to pick up buffer
                        move towards queue of my pick up buffer
                if "other robots within sensor sight with priority = 0"
                        set speed = 0 m/s
                        get "steering angle" of other robot
                        if "side collision"
                                move backwards
                                wait for other robot to pass
                        else ;;frontal collision
                                if "other robot is going to the right"
                                        move to the left
                                else
                                        move to the right
                else
                        set speed = 1.5 m/s
                        move towards queue of pickup buffer
                        if "discent ramp connected to queue of pick up buffer reached"
                                move forward on the ramp
                                if "queue of pickup buffer reached?"
                                        enter the queue
                                        set "priority" = 1
end collision avoidance
```

C.2 Flowchart collision avoidance algorithm

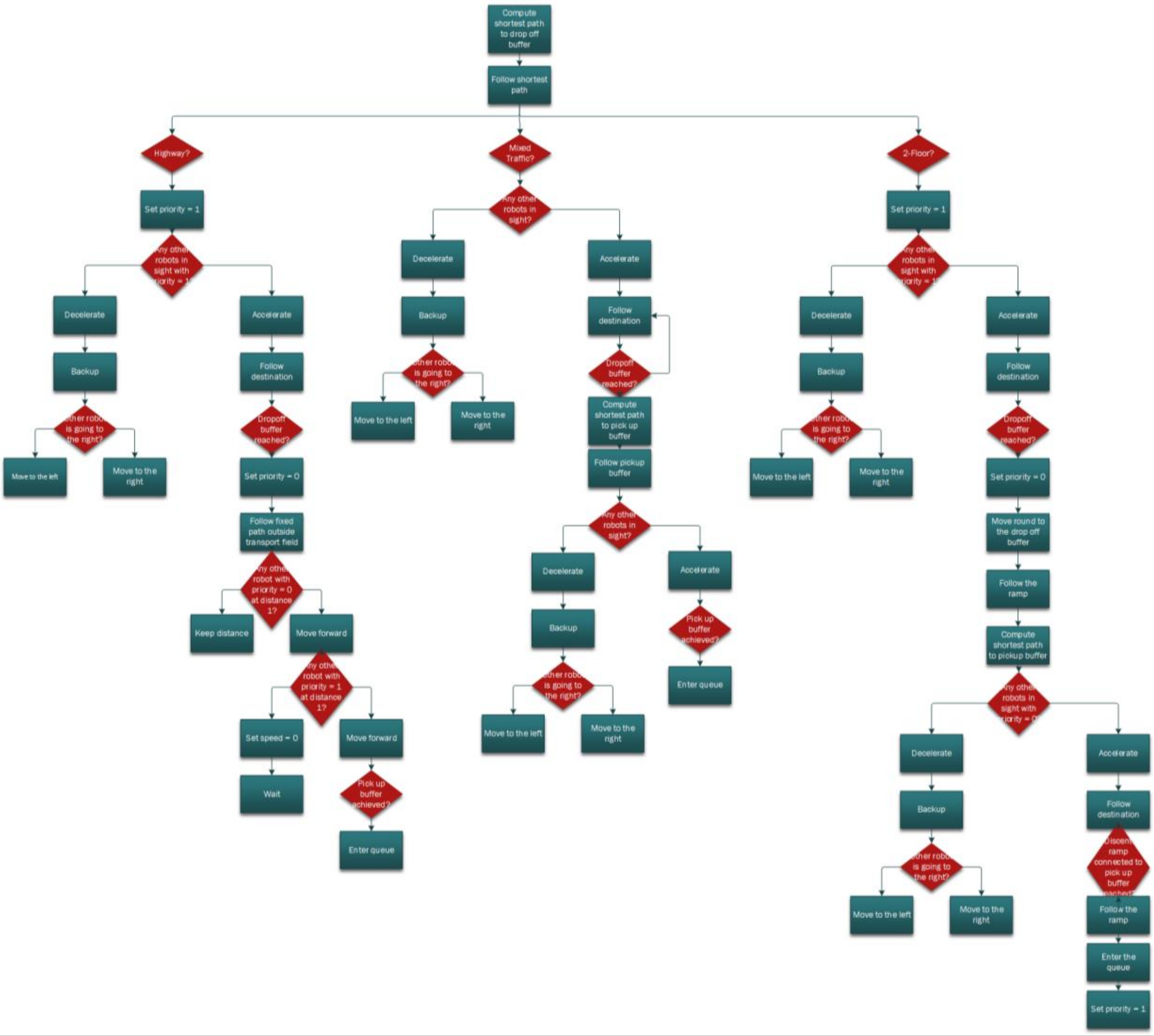


Figure 52: Flowchart collision avoidance

Appendix D

D.1 Pseudo-code individual and cooperative transport algorithm

```
;;Collective or Individual Decision-making

set "master" = nobody
set "myparcel" = nobody
set "loadStatus" = unloaded
set "formation-completed" = false

start decision-making process

if "semaphore1 = 1"
      move to pickup buffer
      if "master = nobody"
           if "not other robots on this buffer"
                      set "master" = agent myself
           else
                      wait
      else
           Ask parcel to set "myrobot" = agent myself
                 set "myparcel" = parcel here
           if "number of robots in pickup buffer = minimum number of robots"
                 set "temporary" = 0
                 do while "temporary < minimum number of robots"
                      if "slave temporary != one of robots here"
                                 set slave temporary = one of robots here
                      else
                                 set "temporary" = temporary + 1
                      end while
            else
                wait for other robots to enter the pick up buffer
                get "code" from myparcel
                if "container code" is situated on the left or in the middle of the warehouse
                      ;;formation pattern 1 for 4 robots
                      Ask slave 1:
                                 set "steering angle" = my "steering angle"
                                 set "lateral position" = my "lateral position + fixed distance"
                                 set "longituadinal position" = my "longitudinal position"
                      Ask slave 2:
                                 set "steering angle" = my "steering angle"
                                 set "lateral position" = my "lateral position"
                                 set "longituadinal position" = my "longitudinal position - fixed
distance"
                      Ask slave 3:
                                 set "steering angle" = my "steering angle"
                                 set "lateral position" = my "lateral position - fixed distance"
                                 set "longituadinal position" = my "longitudinal position - fixed
distance"
           else
                      ;;formation pattern 2 for 4 robots
                      Ask slave 1:
                                 set "steering angle" = my "steering angle"
```

```
                        set "lateral position" = my "lateral position - fixed distance"
                        set "longituadinal position" = my "longitudinal position"
                Ask slave 2:
                        set "steering angle" = my "steering angle"
                        set "lateral position" = my "lateral position"
                        set "longituadinal position" = my "longitudinal position + fixed
distance"
                Ask slave 3:
                        set "steering angle" = my "steering angle"
                        set "lateral position" = my "lateral position - fixed distance"
                        set "longituadinal position" = my "longitudinal position + fixed
distance"
        if "formation-completed = true"
                wait 4 seconds for loading of parcel
                if "time >= 4 seconds"
                        set "loadStatus" = loaded
        else
                set formation-completed = false
        if "loadStatus = loaded"
                follow "container code"
        else
                        wait
else
         wait in queue

if "semaphore2" = 1
        move to pick up buffer
        Ask parcel to set "myrobot" = agent myself
        set "myparcel" = parcel here
        wait 4 seconds for loading of parcel
        get "code" from myparcel
        follow "container code"
else
        wait in queue

end decision-making process
```

D.2 Flowchart individual and cooperative transport algorithm

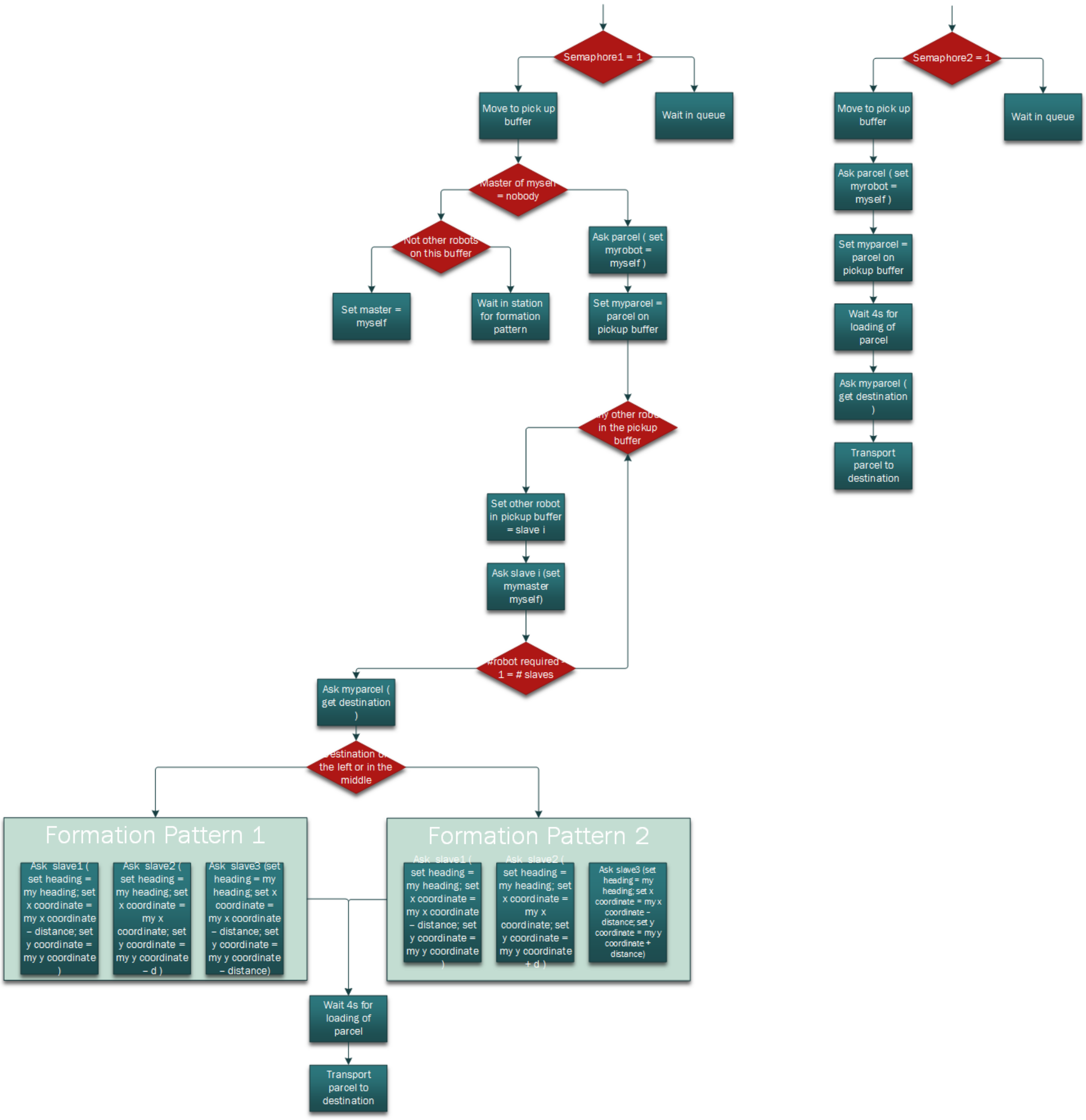


Figure 53: Flowchart cooperative transport algorithm

Appendix E

Postal automation expert interview

Your doctoral dissertation focused on the study of "holonic control for large scale automated logistic systems". In this research, you have defined holons as agents that can make decisions on their own without consulting higher levels of control, but that at the same time need to respond to higher level of control.

"Why can't holons make decisions on their own without higher level of control? Meaning, why not having fully decentralizing systems, where agents or holons can be fully independent from a hierarchical control?"
I agree that the decentralized control has many advantages over a centralized control. During my research, I worked in a project at Schiphol that concerned the designing of an underground connection with automated vehicles. In this project, I strived to have as many decentralized systems as possible. The automated vehicles and the terminals were both decentralized. However, in the eventuality that a fire takes place, you can not use a decentralized system to halt the vehicles and make them follow a certain protocol. Fire is a well-known example that demonstrates some of the disadvantages of having a fully decentralized control. In the project at Schiphol, there was a platform, where containers with flowers were picked up, with two terminals. Moreover, on another side, there was another platform with a big terminal. Thus, if the fire takes place in one of the two platforms, you can still tell the vehicles to stop working in that platform. In addition, since you know in which direction the flame is going, you can tell the vehicles to move to the other platform and continue working. Similar commands are only possible with a centralized system.

"So why don't make it purely centralized?"
The vehicles moving underground decided themselves which packages with flowers to pick up. The decentralization of the vehicles and the terminals made them highly robust and fault-tolerant. With a decentralized control, that system demonstrated to work really good, with high utilization rate of vehicles (over 80%). However, since the vehicles did not have all perfect information, sometimes they did not make the most optimized decisions. For this reason, we decided to have a holonic system. This system had an optimizer on top (central system) that could find better solutions and increase the performances of the system. This is what I mean for holonic approach. I wanted to have as much as possible decentralized, but with a hierarchical organization on top of it. Thanks to the decentralized way, if one of the components of the system did not work, all the others could still work finely. However, for optimization and other cases (e.g. fire), you want to have a top-down control.

"Do you have any experience in the postal market and, in particular, in parcel distribution centers?"
I worked in many parcel distribution centers for DHL, PostNL, Correos (the national postal service in Spain) and Royal Mail. Indeed, I worked for many years for VanDerLande industries. This company produces conveyor systems for postal operators and it is a leader in the sector. I was working in the simulation department, where I had the task of making simulation models out of conveyor belts and the control systems of these machines.

"What are in your opinion the main advantages of using conveyor belts for sorting parcels?"
Conveyors are very reliable, meaning that they make very little mistakes and have very few anomalies. Moreover, they can handle until 10000 to 15000 packages per hour. Therefore, the performance of these systems are very high.

"What are in your opinion the main disadvantages of these systems?"
Conveyor belts are extremely expensive. I visited a parcel sorting center in Kentucky, where the biggest sorter is placed. The cost of this conveyor system was around 1.2 billion dollars. But this was a really huge sorter. The usual investment cost for conveyors is 15 to 25 million €.
The other disadvantage of conveyor systems is that, once installed, you can not do much with them. I did a project for PostNL where they have conveyor systems that consisted of small turning plates and if a parcel is executed, it drops over and falls off in chutes. Those plates were made for small packages. If you had big packages, you needed to have two chutes working together. They never thought about that parcels will grow in size and weight. People buy more often TVs, computers, and other big-sized items online, while before packages were small, with only, for example, CDs inside. The use of two plates for one big parcel reduces the capacity of the system. Moreover, the negative think is that you can not change it anymore the system, because the system is fixed.

"What are the key performance indicators for the postal operators regarding their sorting systems?"
For postal companies, the most important performance indicator is the number of parcels that a sorting machine can do per hour. The number of parcels per hour is of utmost importance for PostNL, considering that it has stipulated an agreement called Universele Postdienst or simply UPD, with the Government. This agreement states that PostNL needs to process about 95% of the parcels per day. If they can not make it, they get a fine.
Another important KPI is how much is leftover at the end of a shift (day). I did a project for the Royal Mail Group (British postal company). They had a problem in their sorting centers that when they closed at night, they had many unsorted parcels left. These parcels were sorted the day after. This means that the parcel, before being delivered to the customer, waited one day. This is very bad for your performance. Empty floors at the end of a day are very important for postal operators.

At this point of the interview, I have illustrated the scope of the research and described the system that I have in mind to solve the problems experienced by postal operators due to the use of long-fixed machineries. Then, I have asked other questions.

"Can a fully decentralized system function adequately in this case?"
In my opinion, drones can be fully decentralized, but they need to respond to a higher level of control in some occasions. For instance, if you have charging stations, then you need to have a centralized control system. This because you do not want the drones to decide on their own when to recharge, since you may incur in a situation where too many drones go to recharge their batteries. Instead, you want that, for example, only 10 drones at the time be recharged. Another problem that need to be tackled by a central communication might be packages that arrive late. Some parcels have higher priority than others because have waited too much in the sorting center. How are you going to handle them in a decentralized way? Other type of parcels are valuable parcels. Some parcels are contained in special containers, which are locked. How are you going to handle them? These problems need to be solved.

“What do you think are the advantages of a system that consists of a fleet of UAVs performing the sorting tasks?”

A clear advantage of your system is that if you have a conveyor belt and the flow of parcels grow, you cannot do anything about it. While, using your system, you can buy additional 20-50 drones. Thus, the system is scalable. From an investment perspective, this can be very appealing.

Furthermore, conveyor belts are usually 10-20 meters high. Your system can be instead only few meters high. Why can’t we then have warehouses with two or more floors. In this way you can double or triple the capacity of a system. This is a bit off-the-shelf thinking but might be interesting to take into consideration. Length is really expensive in the Netherlands, so they might be very interesting in it.

Moreover, your system might be very robust to changes. For instance, in the beginning, you might have 20 light drones that pick up light parcels and 10 bigger drones to pick up heavier parcels. Then, if you have more heavy parcels, you can just buy more bigger drones.

The building are really long and so the distances are really long as well. If you have two floors, you can have much better performances, since the distances are much shorter.

“What problems need to be investigated to show the feasibility of the system by means of simulation?”

The drones are battery-driven and the batteries do not last long, so in the model there should be charging stations. They can fly for instance for half an hour and then they can recharge.

The disadvantage is that if you have 200 of drones and they can fly only for half an hour, that means that you need to charge them four-five times or even more, you need space also for the charging stations.

Moreover, the drones will need time to take a parcel up, so when that delay needs to be implemented. Of course, this might also reduce the capacity of the system.

In the postal market, people are really old-fashion and conservative. They would come up with questions like: your drones fly for 1 m/s, that’s too fast, they will never going to make it. You should do a sensitivity analysis, so not only 1 m/s but also 0.5 m/s. Then, you can say, if the drones go 0.5 m/s we are able to obtain this. However, if you add couple of extra drones you have the same performance. The same you can do for the charging time. They might say 5 minutes are not enough to recharge a battery, it will take 10 minutes or half an hour. You can again show this in your simulation. Show that you have already thought about these problems.

I did the same in my doctoral project. I showed how much the charging time should it be before it would impact the performance of the system. Then the customers could see that even if it took 12 hours to recharge the batteries, the performance was still good. In this way, all the skeptics got an answer. You could say, as long as it stays below 12 hrs, it is fine. Things like those, you can do really well in simulation.

In my opinion, the system is feasible, but you need to keep in mind that the industry is really conservative. They will try to find faults in your system. For instance, two drones together can lift 20 kg, but you also might have parcels of 23 kg. You have to think if that happens, then a solution is ready.

Another thing to take into account is the human-drone interface. Safety systems are required especially for these interfaces. For instance, you do not want to have an operator under a drone, if it has to transport a heavy parcel that might drop on his head. Interactions are mainly in the pick-up stations. You could have a stop-and-go a system. If you push a button the drone stops and when you release that button, it will restart.

I also suggest you having pictures of the entire warehouse and seeing where most of the traffic take place (congestion) and highlight it in red. Then, highlight in green, the areas where there is no congestion, and change the layout until the congestion is

reduced. Move the pick-up stations or the drop-off containers until the congestion is minimized.
Another important thing to show are the organization of special packages, i.e. packages without bar codes or with unreadable bar codes. How are you drones going to handle these packages?

"What are the most important KPIs for this system, in your opinion?"
First, the number of parcels that drones can handle per hour. Then, the leftover at the end of a shift. These are the same for all sorting systems. Furthermore, in your system, other important indicators are: the utilization rate of drones and the utilization rate of the pick-up and drop-off gates.

"How would the layout of the parcel center change with this system?"
If you have bought a new warehouse, you buy it for your pick (e.g. for December when people buy presents for Christmas). In January, you still have the same space and you can't do a lot with it or change it, considering that the conveyors are fixed. Thus, you have space left in your system.
The layout of this system can be very different from the one of the current warehouses. They are really long in size at the moment and as such the distances are very long. With your system, you can have drastically reduce that distance. It can be a small warehouse. Then you can have more things to optimize.
Furthermore, as suggested earlier, it can be a warehouse with two or three floors, in order to double the capacity.

"What data are important to be collected to develop this model?"
Important data are, for example, time that drones can fly with one battery, or maximum distance between drones. Other data concern, how many parcels are needed to be sorted per hour; the percentage of parcels that arrive without bar codes or with ruined bar codes; the weights and dimensions of parcels and others.
You can also play around with numbers. For instance, you can say initially that only 5% of the parcels have a bad bar codes. In another case, you might have 10% of the parcels with unreadable bar codes.
In your doctor research, you stated that: "Attempts to join the fields of simulation and emulation are futile. Simulation is based on the reduction of reality; emulation is based on keeping reality as it is".

"How can be the reality of simulation be enhanced?"
Things like sensitivity analysis and scenario building is what can make your simulation trustworthy.

"Can this system provide a competitive advantage to PostNL?"
Today, there is a lot of competition in parcel market. DHL, PostNL and many others companies are doing the same. A new system like this one can provide PostNL with a great competitive advantage over its competitors.

The interview is over. Thank you very much for your patience and helpfulness!

Appendix F

Design Choices: Retrospective to the main design choices

In this Appendix, we explain why we have made certain design choices, describing what other experiments we have tested and why they did not provide the expected results.

F.1 Developing a map building algorithm

To solve the localization and mapping problem, which deals with the construction of spatial models of physical environments, we have implemented a multi-robot coverage coordination algorithm. The process to develop this algorithm was not linear, with other strategies being experimented before selecting this algorithm. In the initial conceptual model, robots had full awareness of their environment. Therefore, they were able to start performing sorting tasks knowing precisely the positions of the various pick-up and drop-off buffers. Subsequently, we decided to shift from a fully centralized to decentralized schemes to increase the scalability, adaptability and flexibility of the system. For this reason, we wanted to experiment solutions to make the robots themselves performing the mapping of the sorting area. The initial idea was to use the Reynolds's flocking algorithm. Using this decentralized algorithm, robots simulated the behavior of boids of birds, which use three simple techniques (separation-alignment-cohesion) to move in teams. Therefore, when a robot in a boid detected a pick-up or drop-off buffer, it transmitted this information to its flockmates. This algorithm guaranteed very high scalability and flexibility, but poor performance since the mapping time was highly unstable due to the random movement of robots. Moreover, the actions of robots were also highly unpredictable. The second option was to disperse randomly robots in the environment and use randomized gossip algorithms to spread the information acquired by the individual robots. In comparison to the flocking algorithm, this algorithm also guaranteed high scalability and flexibility, but in addition, it provided slightly better performance. In fact, information moved quickly from the first robot acquiring it to the last, but still the performance fluctuated too much considering that, the movement of robots was again random. Finally, looking at the scientific article of D.L. Martínez and A. Halme (2016), we got the inspiration to divide the workplace into four identical portions and place groups of robots in each portion of the space. Robots were place in a circle and they could only move forward. Each group of robots could only move inside one portion. When robots detected a pick-up or drop-off buffer, it transmitted to the other robots within the same portion of the space its findings. Robots receiving the information could then decide to add this information to their list of findings or not, depending on whether this information was already in their possession or not. When all the portions were scanned, each group of robots had fully detected the space. To be sure that every portion of the space was completely scanned, we colored the space (patches) scanned by robots with a white color and we assigned a certain vision angle and radius to the robots. In this way, we were able to calculate precisely how many robots were needed to scan a portion of the space and what vision angle and radius were required. Therefore, when all groups of robots finished scanning their portions of space, they had acquired all information in their portion of space and they exchanged their findings between each other. In this way, robots were able to detect the whole warehouse in a very limited time, increasing enormously the performance, reducing the resources required to do these operations and retaining high scalability and flexibility.

F.2 Developing a resource allocation algorithm

The resource allocation problem refers to the way we allocate robots to the pick-up buffers. As described earlier, this problem has been using a min-max heuristics algorithm, in which robots either choose a strategy that minimizes the maximum waiting time of parcels or select randomly a buffer and follow it. As for the map building algorithm, the process to develop this algorithm was not straightforward and many other strategies were tested that have promoted the use of a heuristics. The first basic idea was to assign robots randomly across the various pick-up buffers. Therefore, every time a robot dropped off a parcel at destination, it chose a random pick-up buffer to follow. This solution guaranteed low computation and good distribution across the buffers. However, robots travelled sometimes very long distances before arriving at pick-up buffers and interfered many times with the motion of other robots. The second strategy consisting in minimizing the travelled distances of robots (shortest-path). This algorithm was again weak, since robots were overburdening some buffers, especially the ones on the sides, while others were visited only very few times. This problem was more accentuated considering that the destinations of parcels are random. Therefore, there were cases in which some buffers were almost not utilized. In addition, in the scenarios with large parcels, robots waited long times to build formations. The subsequent strategy was to let robots communicate with buffers to receive information regarding the number of robots inside the buffers and in the queues. The aim was to improve the distribution of robots across the buffers by exploiting the information on the number of robots in queues and inside the buffers. Considering that robots make the decisions to follow one buffer every time they drop off a parcel at destination, multiple robots followed the same buffers that in that instant had the lowest number of robots. Therefore, buffers with the lowest number of robots were suddenly overburdened with robots. To solve this problem, we let robots communicate several times with the buffers to update the number of robots of queues that could not exceed a maximum threshold. However, in this case, robots sometimes travelled extremely long distances before finding a buffer, while other times they changed multiple times their destinations. Therefore, we finally decided to use a heuristics to counteract all these identified problems. The idea was to maintain a certain degree of randomness in the decision-making process of robots to eliminate the excessive allocation of robots to certain buffers. The strategy was then to assign a probability to the decision of robots to follow a buffer. Following the study of A. Farinelli et al. (2017) we have decided to experiment a min-max heuristics in which the decision was to either ensure every buffer had at least a robot and keeping some randomness in the decision. In this way, we were able to distribute effectively robots across the pick-up buffers. However, we were not still fully satisfied about the time robots took to build their formations. Therefore, we have exploited another information available, which is the waiting time of parcels at pick-up buffers. In fact, ensuring that every buffer has a robot is not enough considering that buffers with big parcels necessitate more robots than others. We knew from the statistics already in place that big parcels had larger waiting time in comparison to small parcels, therefore robots could exploit this information to reduce the time to build formations. This entails that more robots should be directed towards buffers with big parcels than to buffers with small parcels. To do so, robots should try to minimize the maximum waiting time of parcels, which is highly likely to correspond to a big parcel. However, to avoid robots moving all together towards the parcel with the highest waiting time, we decided to keep some randomness in their decisions. Therefore, we assign a certain probability p that robots move towards the parcels with highest waiting time and probability *non p* that robots move towards a random station. Finally, to reduce the travelled distance of robots, we decided to complete this algorithm with the insertion of a maximum distance robots can travel to reach a buffer. The calculation of the maximum distance, shown earlier, also intends to reduce the interferences among robots.

F.3 Developing a task allocation & motion coordination algorithm

Task allocation refers to the allocation of parcels to robots. In this situation, this is also related to the formation-building problem for robots that need to transport heavy and high volume parcels. As already argued, to address this problem, we have used a dynamic behavior switch of robots together with a combination of a leader-follower and auction-like algorithm. Before designing this solution, other strategies were tested. In the beginning, we desired to develop a fully decentralized strategy for the transport of big parcels. Therefore, we developed an artificial potential field approach, in which all robots were aware of the destination of the parcels and followed this destination. While moving, robots tried to maintain a certain distance from the gravity center of the group, thus using repulsion and attraction forces to stay close enough but not too close from that point. This solution could work if there were no other obstacles in the environment. However, considering the presence of other robots that impede the straight trajectories of robots, the formation did not maintain a good stability. The other fully decentralized strategy used in previous studies for the cooperative transport of objects lies upon the behavioral approach. Using this strategy, agents coordinate their movements through local perception and indirect communication (stigmercy). However, this approach provides poor quality solutions and does not guarantee stability to the formation. Therefore, we have decided not to pursue with fully decentralized strategies, but to test strategies that would not take considerable time to coordinate the movement of robots and that would provide sufficient stability to the material transported also during collision-avoidance. Hence, we decided that the only two solutions that seemed appropriate for the coordinative transport of big parcels were the creation of virtual structures and the use of leader-follower algorithms. After an extensive study of the literature, we have decided to use a leader-follower strategy, which in comparison to the virtual structure approach requires lower inter-robot communication and higher scalability. Obtaining higher scalability for the task of transporting big parcels is not essential, but it is a convenient attribute in case we desire to change the formation in size and shape. For instance, if we notice that the parcel does not have enough stability, it is possible to ask robots to increase or reduce their distances or to add another member. This is not easy to obtain using a virtual structure approach. Moreover, both approaches are centralized by nature; hence, both lead to deteriorating the robustness of the system. For the building of formations, a considered plan entailed the division of robots into two categories, namely robots able to cooperate and robots not able to cooperate. This resembles a set partitioning strategy, where sets of robots are formed that are assigned to specific tasks. In that case, a set of robots could have been assigned to tasks that required cooperation, whereas the remaining to the tasks that required no-cooperation. This solution provides an easy resolution to the task assignment problem, but it requires the use of heterogeneous robots and it could not work well in all scenarios (e.g. in case of only small parcels or only big parcels). Therefore, we have lastly decided to implement a switch behavior algorithm in which the same robots could perform different types of task. In addition, the research of J. Guerrero and G. Oliver (2012) inspired us to use this algorithm in combination with an auction-like algorithm and leader-follower for the task assignment and building of formations of robots. This solution can provide an optimal usage of resources, with homogenous robots being able to perform various tasks, high scalability, high formation-stability and (using also the min-max heuristics) low time to build formations.

The recruiting process involved all robots without parcels. However, this procedure guaranteed low quality solutions, since sometimes robots at higher distance were preferred over robots at short radius. With the introduction of queues and the communication robot-to-pickup (for the open/closed state), leader robots could recruit other robots within a short distance, stepping up the formation-building process. To eliminate potential deadlocks, big parcels were only assigned to the leaders according to a 1-to-1 assignment. Regarding the collision avoidance, we have noticed that using the sensing data of one robot (leader) is not enough and, consequently, some collisions were reported. Therefore, we have decided to expand the vision angle and of the leader exploiting the data coming from the follower placed on its side. In this way, collisions were appropriately prevented. To increase further the situational awareness of the leader, it was also observed that changing its position within the formation based on the position of the destination was a convenient method. Therefore, we have developed two possible formations, where leaders can interchangeably switch their positions.

F.4 Developing a path planning & collision avoidance algorithm

Path planning concerns with the selection of a route to transport parcels. In this specific problem, robots need to find safe collision-free paths. Therefore, path planning and collision avoidance need to be studied together. In this case, we have decided to use a behavioral-based approach, as proposed by D. Sun et al. (2014), in which robots decide to follow a trajectory independently, i.e. without analyzing the paths selected by other robots, and deviate from their paths only in the proximity of potential collisions. This swarm intelligence approach is deemed able to provide high scalability and low computational time to calculate safe paths. Moreover, we wanted to prevent the dependence on grid like structures (seen in the previously implemented multi-robot systems in warehouses) to increase the flexibility and adaptability of our system. Therefore, we have opted for a path planning method different from the A* that does not allow searching in every angle. NetLogo has already a built-in path planner that resembles a Basic Theta* or Phi* (A* variants) that allow moving in every angle. Thus, we have exploited this potentiality of this simulation software program. Another idea was to use the A* algorithm and divide the patches into triangles and make robots move diagonally other than laterally. However, using NetLogo, there was no need to do this. Other strategies that we evaluated consider the use of pre-defined routes from pick-up to drop-off buffers (centralized approach) or the use of pre-sorting to place parcels at known pick-up buffers and predict the trajectories of robots that could have then been synchronized (e.g. wave-movement of robots). The former approach does not provide scalability and flexibility, while the latter required expensive technologies to perform pre-sorting operations. Another alternative implied the use of a market-based algorithm to negotiate the trajectories of robots. However, this strategies necessitate the use of a central agent (auctioneer) thus decreasing the robustness of the system. Eventually, we have observed that using the information about the steering angles of robots, robots were able to understand and handle conflicting (side or frontal) trajectories. This algorithm is simple, scalable and guarantees high flexibility. In certain traffic configurations, we have revised the algorithm with the inclusion of priority schemes. For instance, in the highway, after the last reference points were passed, robots without parcels impeded robots with parcels to leave, reducing the speed of the transport operations. Therefore, we have decided to assign lower priority to robots without parcels and give higher priority to robots with parcels. In this way, robots with parcels do not have to wait or dodge robots while leaving the exit queues. In addition, in this configuration, we have initially designed the lateral paths as fixed, with robots not having the possibility to overtake other robots. However, this was a big limitation considering that if a robot fails in that line, all the preceding robots would be stuck.

Therefore, we are using reference points instead with robots having the ability to overtake each other. Nevertheless, overtaking is not necessary unless there are failed robots as observed in the simulations.

F.5 Choice of traffic design alternatives

Different traffic design alternatives can be employed to control the traffic flow inside the sorting center. In the beginning, we were only considering one traffic design option, being *mixed traffic*. Clearly, this design alternative provided high congestion in the area. Therefore, we have decided to design additional alternatives that were able to reduce this congestion, thus enabling higher safety for the sorting system. Looking at the various transportation approaches used to direct the traffic of vehicles, we have decided to implement two additional solutions: *highway* and *two-layered*. The highway consider the separation of robots moving in opposite directions on the same plane. This alternative resembles the way the traffic is managed in a traditional highway, where vehicles moving in opposite directions are separated on different lanes. For the configuration of this traffic alternative, we explored multiple options. The initial idea was to construct two queues linked to the drop-off buffers, one entry queue and one exit queue. The exit queue was then linked to a route guiding robots outside the transport field. However, in this configuration, many interferences between robots leaving and entering queues were left over. Therefore, this did not provide the results we wanted to achieve with this configuration. Another plan was to let the fixed path cross the transport field to reduce the travelled distance of robots. Again, this idea was discarded due to the high interferences left among robots travelling on opposite directions. Eventually, we have opted for the designed configuration, in which robots arriving at the drop-off buffers, follow the lanes outside the transport field, thus eliminating the interferences between robots moving in opposite trajectories. As shown in the results from the experimental designs, this solution guarantees very high safety and high predictability of the motion of robots but it also induces robots to travel long distances before arriving at the pick-up buffers. In order to obtain a solution that provides high safety and reduces the distances travelled idle of robots, we have designed a third option, being two-layered. This traffic design alternative produces optimal results in terms of system effectiveness and safety. Therefore, it represents a good compromise between the first and the second option already analyzed. However, this configuration might decrease the flexibility and adaptability of the system in case UGVs are used. Therefore, the first two options might be preferred over this one when using ground robots, while two-layered should be tested with the use of UAVs.

F.6 Design of the simulation environment

In Chapter 2, we have showed that sorting terminals can have different shapes, e.g. rectangular, cross-shaped or U-shaped, according to the configuration of the sorting machine adopted. In our model, we have opted for the most common configuration, being the rectangular sorting terminal. However, given the adaptability of the system, it should not a problem to test it in other terminals. The dimensions of the sorting space was decided after a calculation of the space needed by robots to move with some degrees of freedom and to place enough pick-up and drop-off buffers. In every sorting center, the number of pick-up buffers is lower than the number of drop-off buffers (almost 3 times lower). Therefore, we have decided to place 20 pick-up buffers and 50 drop-off buffers, which corresponds to a mid-size sorting terminal. Moreover, the research of C.J. Hazard et al. (2006) inspired us to design our system. In the Alphabet Soup, robots are also used to sort items (letter). Therefore, we could use this system as a benchmark for the design of our sorting environment. In that case, however, the number of entry and exit gates were equal. However, as A. Farinelli et al. (2017) also remarks, in most warehouses, entry gates are generally lower than exit gates. Finally, the choice of linking the pick-up buffers with two queues was also done in a second moment. In the beginning, we linked these buffers with one queue placed behind pick-up buffers. However, we have decided to move this queue in front of pick-up buffers to reduce the workspace and reduce the distance travelled of robots. To avoid interferences among entry and exit robots, we have placed to queues, one entry queue and one exit queue. Regarding the dimensions of robots and buffers, we have used the requirements provided by the Prime Vision.

Appendix G

Verification tests

Table 16: Verification tests

N°	Verification type	Agent	Description	Expected Result	Obtained Result	Verified?	Solved?
1	SAT	R	Testing if a randomly selected robot memorizes correctly the pick-up buffer when in sight	Robot 81 will save the pick-up buffer 13 as soon as this is in sight	Robot 81 correctly memorizes buffer 13 in sight and shares its discovery with the other robots	Ok	-
2	MAT	R	Testing if all robots memorize all pick-up and drop-off buffers once the assigned area is covered by their sensors. I will do this by assigning two counters to the robots that are updated every time a buffer is allocated to the memory	Counter = 20 = nr of pick-up buffers Counter1 = 50 = nr of drop-off buffers	Counter = 20 and Counter1 = 50	Ok	-
3	SAT	R	Testing the behavior of a randomly selected robot in a queue	Robot 119 assumes initially the last position in a queue. I want to assess if it moves by one position every time another robot enters the buffer area	Robot 119 moves correctly in a queue. The robot waits in position until the robot ahead moves to the next position in the queue, then it moves to the position occupied by the robot ahead	Ok	-
4	SAT	R	Testing the waiting behavior of a randomly selected robot inside a pick-up buffer loaded with a small parcel	Robot 170 has to wait 4 seconds since the exact moment it arrives at the station. The robot has a variable called ticks-since-here that save the exact second (not tick) the robot enters the areas. 4 s pass after this variable value.	Robot 170 updates its variable ticks-since-here to 843 s when it enters the pick-up station. Then, it waits until around 847 s to leave. Sometimes, the robot will leave after 5 s but this error is admissible in this case	Ok	-
5	MAT	R	Follow a number X of robots (including slaves and masters) dropping of the parcels and see if they correctly follow their assigned path	Robots 96-133-157-116 are transporting a heavy parcel to destination. I am expecting that after dropping off their parcel, they will all follow the highway	The expectation is not respected since only the master of the formation will follow the highway, whereas the slaves will go directly to	No	Yes

					the next pick up buffer		
6	SAT	R	Follow a randomly selected robot and check the decision it makes when a parcel of different weight and size is met	Robot 85 will not change its master or slave status, initially set to nobody, when encountering a small parcel. Vice versa, it will become a master or mymaster when meeting a big parcel	Robot 85 has kept its status to nobody when it was loaded with a small parcel, while it has updated its status to mymaster when helping another master 181 to transport the heavy parcel	Ok	-
7	SAT	PB	I want to evaluate the behavior of a pick-up buffer with regard to the parameter parcel-true?	Location 9 should switch the binary value of parcel-here? to true every time a parcel appears on that location. While the value should be set to false when there is not parcel on it.	The test is performed until 1000 ticks. Location 9 behaves in the right way, switching the value of parcel-here? to true when a parcel is on it and false when a parcel is not there	Ok	-
8	SAT	R	Testing the waiting behavior of a randomly selected robot inside a pick-up buffer loaded with a heavy parcel	Robot 104 is a master of a formation. Once it finds the slaves, it should wait 4 s to give time to the human operator to load a parcel on it	The experiment is not verified. The robot does not count the time it is waiting on the location since it has recruited all the slaves. Correction is required.	No	Yes
9	SAT	PB	I want to evaluate the behavior of a pick-up buffer with regard to the parameter counter2, which is used to count how many robots are waiting in the queue	Location 15 should update the counter according to the number of robots that are placed in the queue	The test is performed until 800 ticks. Location 15 does not reset the counter to zero when there are no robots in the queue. However, when there are robots, the location updates correctly the counter.	No	Yes
10	IT	R	I want to test the collision avoidance behavior of different robots. Robots should be able to avoid each other when a certain distance is not respected.	Robot 133 and Robot 75 have conflicting trajectories. I want to test if they avoid correctly each other, respecting the distance.	The collision is avoided according to the conceptual model, the robots are also able to respect the right distance of 3 patches	Ok	-
11	IT	P	I want to test the interactions between some robots and parcels. Every parcel can be assigned to only one robot (in case of heavy parcel to the master). Once	Parcel 194 should pair up with the first robot in the queue. This parcel thus set its own variable myRobot to the Robot to which is assigned. This	Parcel 194 is paired up with Robot 73, which is the first in the queue. The variable myRobot, thus become correctly Robot 73, and it doesn't	Ok	-

			parcels-robots are paired together they can't change to another robot or another parcel.	parameter is observed and can't change during the process. Same test is carried out for parcels 274, 255, 248	change its value. Therefore, also when the parcel is dropped off at the station, it still keeps the serial number of the Robot that has transported it. Parcel 274 is paired up with Robot 85. The test is successful. Parcel 255 is paired up with Robot 125. The test is again successful. Parcel 248 is paired up with Robot 124. The test is again successful.		
12	SAT	R	There might be a situation where two robots arrive at the same time at the same location considering that 4 robots can access a location with heavy parcels. To check whether the implemented model respond well in these situations, I will test many robots becoming master and evaluate their behavior	Many robots facing heavy parcels will be tested, to ensure that in any situation the robots behave correctly. Multiple runs will be carried out, each with a time of 10,000 ticks.	The robots do not behave as desired and described in the conceptual model. When 2 or more robots arrive at the same time at the same location, usually two robots become both masters. The test is not successful.	No	Yes
13	MAT	R	When another robot is in sensor sight, the other robot should avoid collision. When the potential collision is a side collision, one robot has to wait and back up while the other keeps moving forwards. By contrast, when the potential collision is a frontal collision, one robot should move on the left and the other on the right so as to go round the obstacle	Many robot-robot collision situations are examined. In the end, zero collisions are expected between robots. Furthermore, the side collisions should be easily avoided, while frontal collisions require a bit more time, since the robots should move in opposite directions before moving again. In this test, the number of robots is equal to 100.	The simulation is ran for 10000, and a counter is used to count the number of robots sharing the same patch at the same time. The number of collisions avoided is very high, which shows a good validity of the model. Some collisions still may occur in the beginning of the simulation, where most of the congestion is placed, or on the left side of the highway. This occurs because	No	Yes, the collisions at the beginning of the simulation are solved by making the robots wait more time in the pickup buffer initially, whereas those in the highway are solved by

					on the left side the robots turn right instead of left to avoid other robots. In this test, the number of robots is equal to 100.		imposing the robots to turn left on the highway
14	MAT	R	Same experiment as above, but this time we increase the speed, to see if the robots collide when the speed is too high. In this test, the number of robots is equal to 100.	Increasing the speed might cause more collision to occur. We want to evaluate at which speed the robots start colliding. In this test, the number of robots is equal to 100. I expect that there are no problems with increasing the speed within a certain limit. After this limit, collisions or disruption situations occur. Changing speed parameters: Speed = 0.1 Speed = 0.3 Speed = 0.5 Speed = 1 All tests are carried out for 5000 ticks	Speed = 0.1 : no collisions Speed = 0.3 : no collisions Speed = 0.5 : no collisions in the transport field, but 9 collisions occurred in the highway Speed = 1: deadlock situation, the system does not work when the robots go so fast	Ok	-
15	MAT	R	Same experiment as number 13, but in this case we want to see the effect on collisions of an increase of the number of robots in the field. Like for speed, increasing the number of robots and keeping the size of the area the same might create problems with regard to congestion and thus to collisions. I expect that the system can withstand a certain amount of robots before reaching a deadlock situation and increasing the number of collisions	Changing number of robots: #robots = 150 #robots = 200 #robots = 250 … till reaching deadlock Experiments carried out for 5000 ticks, or until a deadlock occurs. NOTE: this experiment is tried on the model that considers the highway scenario. One clear conclusion that can be already made is that in the mixed traffic a deadlock situation would occur already with less robots. Thus, the highway and 2-floor scenarios can	#robots = 150: some collisions in the beginning. This can be easily prevented when not all robots start at the same time. The system starts working well after initial congested situation. #robots = 200: same as with 150. Collisions may occur initially, where the congestion level is very high. Subsequently, the system starts working well with no congestion in the transport field, but some in the highway (due to size of highway)	Ok	

				support a higher number of robots.	#robots = 250: deadlock situation reached and many congestions occurred in the beginning of the simulation. This entails that in this configuration 250 robots are too many		
16	IT	R	In this experiment, we want to assess how the collision avoidance with priorities work. In the highway scenario, the robots moving on a highway, thus unloaded, have less priority than loaded robots. When unloaded robots arrive to the last reference point, a mixed traffic situation occur where unloaded robots encounter loaded robots. In these situations, unloaded robots have to wait (they have less priority) and let the loaded robots pass first.	We will test many interaction in the last part of the highway between robots with priority = 0 and robots with priority = 1. Robots with higher priority have to go before those with lower priority, which have to wait until the other robot enters the transport field. The test is again ran for 5000 ticks.	Robot 107, with priority = 0, meets Robot 139 with priority = 1. Robot 107 correctly waits until Robot 139 moves to the next patch before moving forward. Robot 86, with priority = 0, meets Robot 102 with priority = 1. Robot 86 correctly waits until Robot 102 moves to the next patch before moving forward. Robot 89, with priority = 0, meets Robots 114, 116, 135, 137 with priority 1 which are transporting a heavy parcel. Robot 89 correctly waits until the other Robots arrives to the transport field, before moving forward. Robots with priority = 1 do not stop their motion when meeting other Robots. This is desirable considering that they have higher priority.	Ok	
17	RTAB	R	Semaphore 1 and Semaphore 2 are two binary variables that allow a robot to enter a pick up buffer when	We want to test if semaphore 1 and 2 switch to 0 when a robot cannot enter a pick up buffer and	Once arrived to the first position of the queue, Robot 104 has updated the value of	Ok	-

			there are no other robots or other robots are required to lift a heavy parcel. Here, we want to test the functioning of these variables.	to 1 when the robot can enter the pick up buffer. These semaphores have then to be again switched off when robots arrive at the drop off buffer and release their parcel. Several robots will be individually tested.	semaphore2 to 1, and thus it has entered the buffer. When it arrives at drop off 66, parcel 216 is released and semaphore2 switches to 0. Robots 130, 111, 158, 77 have updated the value of semaphore1 to 1 and have entered the pick up buffer, containing a heavy parcel. When they arrive at drop off buffer 34, they have released the parcel, and updated the value of semaphore1 to 0.		
18	RTAB	P	Every parcel has a waiting time, which counts how many seconds it waits before being picked up by a robot. In this experiment, we want to assess if the value of waiting time is measured correctly.	We will examine a number of parcels to see how they update their waiting time value. In the beginning of the shift, the waiting time is expected to be very low considering that, the robots are already positioned in the queue. Subsequently, I expect that this value will increase more. A plot is made to check these statistics.	In the beginning, the average waiting time of parcels is very low, but after some time, the time increases rapidly. The parcels correctly update the value of waiting time. Initially this is set to the seconds in which they appear on the pick up buffer. This value is updated to right waiting time once the parcel is claimed by a robot.	Ok	-
19	RTAB	P	Every parcel has a service time, which counts the seconds it takes a robot to transport the parcel from its origin to destination. In this experiment, we want to assess if the value of service time is measured correctly.	We will examine a number of parcels to see how they update their service time value. The value of service time is expected to increase when there is more congestion in the area, so in the beginning of the shift, and to lower after the initial congestion. A plot is made to check these statistics.	The service time is measured correctly. Initially, the value is set equal to the time (in seconds) in which a robot has claimed the parcel. Once parcels arrive at destination, the current timer value is subtracted to the previous value. The service time of heavy parcel is logically higher than that of light parcel, considering the time a robot takes to	Ok	-

					recruit other slaves and the fact that they move slower than robots loaded with light parcels. The service time increases with the amount of congestion in the area.		
20	SAT	P	Every parcel appear on a pick up buffer when a robot has taken the previous parcel. The parcel then moves on a robot when the robot claims it. Finally, when the robot arrives at destination, the parcel is moved to the drop off buffer.	In this experiment, I want to visually assess whether a parcel moves from a pick up on a robot and from a robot to the drop off buffer.	Parcel 431 arrived at pick up buffer, then when Robot 132 claimed it, the parcel moved on the Robot. Eventually, when Robot 132 arrived at drop off buffer 21, the parcel moved onto the container. Parcel 377 is a heavy parcel. It moved on Robots 74, 137, 100, 71. When arrived on drop off buffer 24, it moved onto the buffer. The experiment is carried visualized on other Robots, and all tests were successful.	Ok	-

Appendix H

Validation test 1: Interview logistics expert

The interviewee is a full professor at the transportation department of the Faculty of Technology and Management of TU Delft. This professor is an expert of logistics and freight and his interests also concern the areas of agent technology, process optimization, collective behaviors and self-organization.
The interview is structured in a way that the questions can come from both parties. In the beginning, the thesis project is illustrated, with an explanation of the research questions, the process diagram, the algorithms used, the KPIs employed and the scenarios that we want to investigate. After this brief presentation, the different models are displayed. Subsequently, the interview starts.

1. ***To what extent does the model represent a real-life sorting operations?***

This system is still in an exploratory stage, however the sorting operations are realistic with robots picking up parcels at appropriate points and transporting them to the required containers. The configuration is also realistic, since in sorting centers there are typically more exit points than entry points. Regarding the percentage of small and big parcels, I suppose you have checked this with the data from PostNL. Therefore, the sorting operations performed by this new system are indeed representative of a real-life sorting situation.

2. ***Considering the objectives of this research, do you believe that the model's outcome can provide answers to the investigated research questions?***

    - ***Are the employed KPIs adequate to make comparisons between scenarios?***

The KPIs utilized are adequate and suitable to make comparisons between scenarios. Nevertheless, you should also state why maintenance-related indicators are not taken into account. In fact, a maintenance planner would like to know when robots require maintenance in order to make an optimized schedule. This is not essential at operational level, but it is more important at the tactical-strategic level. The same can be stated for energy problems. If you are not considering scheduling problems, which are indeed at different levels than operational, you should make this clear.
Furthermore, performance generally consists of service/quality and cost. In this project, you are considering only the service/quality part, leaving out the cost dimension. However, most of the times, when dealing with logistics problems, the objective is to minimize costs given certain service/quality constraints. In my opinion, you should use another term in place of performance. You can use effectiveness or service/quality to collocate better your analysis.
The way you show congestion is also a bit disorienting. In transportation science, congestion has a negative connotation, and in the way you calculate it, the more it increases the better. You should make it the other way round, in order to see the opposite trend.

    - ***Can the model show if the MRS is robust?***

The way you are addressing the robustness problem is neat, since the robustness of a system is indeed demonstrated by the ability of the system to recover from failures or to keep working with good service. The recovery mechanism that you have shown in the model, with other robots placed outside the field, entering to help the failed ones gives an accurate overview of how the problem can be solved. Robustness of primary operations is different from maintenance, which is at tactical-strategic level. You should show the recovery system in one paragraph of you thesis. It is worthwhile to show, as it gives an extra value to your research.

- ***Can the model show if the MRS is flexible?***

The flexibility of this system is shown by the fact that robots are able to cope with changing parameters, like for instance the percentage of small and big parcels. Another way to address flexibility is to see how the system works in other layouts, so changing the position of the structural elements.

- ***Can the model show if the MRS is performing well?***

Your model can show if the system performs well in terms on how many parcels are transported by robots per unit of time. However, your model can not show the economic feasibility of this system. So performance should be rephrased in service or effectiveness.
Furthermore, in order to prove if the system works with good performance, you should make comparisons with the currently used sorting machines, using the data provided by PostNL.

3. ***What is your opinion about the analyzed scenarios? Can they adequately address the research questions?***

Your "traffic" alternatives (his suggestion was to call them not system alternatives but traffic alternatives) are systematic, considering that you have robots mixed on the same plane, separated on the same plane and separated on different planes. I do not see any other alternatives to these.
I would like to understand why you gave so much emphasis on congestion. This is a very interesting phenomenon to analyze, but does it really affect the service time of robots?
The answer was that congestion could be used to address the workspace required in the warehouse to have such a big number of robots. Furthermore, the less the congestion, the less time wasted by robots to avoid each other, thus having a relative impact on service time.
I imagine it influences a lot which algorithm you are using for collision avoidance. If you pre-plan the collision avoidance, having fixed paths for robots, you might not have collision at all. You could optimize the set-up paths at higher level, to minimize the minimum time of collision and diversions.
The answer was that we are considering the robots as entities having a control authority on their own, thus making them less dependent on centralized schemes. Furthermore, we want to make this system workable in every situation, e.g. any warehouse configuration. Optimizing the system for a specific layout reduces the reusability of the system for other layouts. There are other algorithms for path planning and collision avoidance that may give better results, but we tried to make the robots as simple as possible, therefore less intelligent with regard to the possibility to anticipate the behaviors of all other robots.
This is of course more advanced control theory, but you should put in your recommendations that better solutions may exist.

4. ***What other results/experimental designs or other robots' behaviors do you think should be implemented in order to fully answer the above-mentioned questions?***

In my opinion, it would be interesting to implement the learning behavior of robots because robots can learn socially and individually. They might improve system effectiveness by doing so. Furthermore, you could add different path planning and collision avoidance algorithms to reduce the number of diversions and time wasted. Looking at different modes of recovery would also be nice. In addition, scheduling optimization of maintenance and energy-related problems can be used to further improve the model.
All these can be written in the end as recommendations for further studies on top of your research.

5. ***Do you have any other questions concerning the validity of the model?***

Yes, I was wondering why the highway is not made for loaded robots, but just unloaded robots?
Answer: This is done in consideration of the fact that every parcel has a unique code, so we do not know which parcel ends up in which station. Due to this, you may incur in many situations where robots travel very long distances before arriving at the parcel destination. Furthermore, you would have more robot-to-robot interferences, considering that many robots would go in different directions on the path. This may lead to a higher number of collisions also with the unloaded robots entering the queue. A pre-sorting could be made so as to make sure that parcels arrive at the pick up stations that we desire, but this technology would be too expensive and it is thus left out from additional considerations.
Is the path in the highway in 1D or 2D? Meaning, can the robots avoid other robots in the highway, taking over other robots etcetera?
Answer: The Highway is implemented in such a way that a robot can overtake another teammate; however, there is no need to do it, because they travel at same high speed on the highway, so you can merely see any overtakes. In the disruption scenarios, however, more overtakes are visible, since the robots have to take over failed robots on the highway.
Why is the utilization rate really high in the beginning and it drastically goes down after some time?
Answer: This is due to the fact that robots are initially placed in queues and they are soon ready to take a parcel. After all of them take a parcel, the utilization rate drastically drops before levelling up at a certain point. This could be changed by either making the robots leaving the stations in different waves, or by taking results into account only after X number of seconds.

Concluding, you have conducted a very nice and thorough research. Of course this is a new ground, so present very formally the results of your simulations. You can create an instance, which can be used by others to test other algorithms. If you show explicitly, what your parameters are and what conditions you have changed, other researchers can compare their work with yours. Show everything you got. Make sure that all the interesting aspects result from the thesis.

Appendix I

Validation test 2: Interview simulation and automation expert

1. ***To what extent does the model represent real-life sorting operations?***

It is not important whether the system represents the current real-life sorting operations, considering that these operations may be adapted to the multi-robot systems. From my perspective, you should forget existing systems, because this system provides a level of flexibility that it is impossible for existing systems to reproduce.

The models you have developed represent to a great degree the sorting operations. However, there are some omitted aspects in these models. For instance, you have not taken into account the most important KPI for a sorting hub, being the compliance with the SLA (service level agreement). PostNL has an SLA with the government according to which this postal operator must deliver 99% of all packages to the customers within 24 hours. In these sorting hubs, the first truck arrives between 5 and 7 am and it has to leave at 11 am with all sorted parcels to go to another sorting center. If this truck has brought 20000 parcels, for example, you need to sort within the interval time all the parcels. The truck cannot wait longer time; otherwise, PostNL cannot comply with the law and will pay a stringent fine. Therefore, your system can have very high utilization rate and throughput and so on, but the most important KPI is the compliance with the agreement with the Dutch Government. In order to do this, however, you would need many data, such as the arrival patterns of trucks, the hourly demands etcetera. Thus, I can understand that you cannot consider all these aspects, but for a real-life situation, this is of extreme importance. Furthermore, you have assumed that all incoming parcels arrive continuously at the pick-up stations, with perfect distribution. This is not the case in real sorting situations. In reality, three pick-up stations are typically connected to one inbound truck. Some trucks arrive earlier than other, meaning that robots can only get parcels from those stations. Subsequently, other stations open and they can move towards the others. The demand of pick-up stations also fluctuate within a single shift. For example, at a certain point a truck arrives with 12000 parcels, while in the next hour another truck arrives with 6000 parcels and so on. Therefore, the demand is not constant. How do you deal with these situations?

Answer: A decision-maker can use the performance indicator of robots; in terms of how many parcels can a robot deliver per hour. Given the scalability of the system, if you have higher or lower demand you can increase the number of robots depending on how many parcels you need to sort. Regarding the fact that some stations can be initially closed, we can face these situations exploiting the potentiality of the min-max allocation algorithm, where robots try to minimize the maximum waiting time. If more parcels are at few stations or only at certain stations, robots will still strive to minimize the waiting time which will be higher for parcels at these stations.

If you make some experiments where you give an uneven distribution of packages at the pick-up stations, you can also see how robust your algorithm is. Increase by 3 or 4 times the amount of parcels at some stations to show the robustness of your system.

Another important aspect is missing in your model that regards a potential disruption of the system. The big advantage of conveyors is that they have an energy supply that can be plugged in a socket and the system does not need recharging. When you have robots, the batteries need to be recharged or substituted. This recharging process may take 5-10 minutes per robot. If you have 200 robots, this can cause serious disruption.

This should be also taken into account, even with some assumptions. Maintenance is another point, but it is less frequent than energy recharging, thus, does not need to be considered for now.
Furthermore, some parcels might have higher priority than others because they have to be delivered within 24 hours, while others maybe in 48 hours. This should be also implemented in the model.
What you need to do is to look at the side-effects or limitations of your system and clearly indicate them. These considerations can also be made as recommendations for future work.

2. ***Considering the objectives of this research, do you believe that the model's outcome can provide answers to the investigated research questions?***

   - ***Are the employed KPIs adequate to make comparisons between scenarios?***

All the KPIs are good to compare the different scenarios, but some other KPIs should be added. Apart from the SLA, for which you would need an extensive number of data, the other KPI that you should implement is the distance travelled by a parcel and compare it with the distance travelled by a robot. These distances are not the same, because robots have to travel with parcels but also without parcels. You should make a ratio between these two figures and see the results between the scenarios. Service time gives already an indication of this, but remember that distance and time are very different. For instance, when a robot avoids a collision, this consumes time but not distance. Therefore, service time can go up, but travelled distance remains the same. If the ratio is low, it is better because it entails that there are less robots idle and, also, that the energy consumption is probably less. This indicator is also helpful to further validate your dispatching or resource allocation algorithm. I expect that if you have more robots, the ratio of distance travelled would be reduced, and it would go below 50%. The closer you get to 50% the better.
Furthermore, service time, distance travelled and also congestion are very good indicators to estimate the most appropriate number of robots to use, to find the turning point where you can say I have enough robots now. You should make some graphs to show the optimal number of robots per different traffic configuration.
Using congestion is also good, because a decision-maker can decide whether he wants higher throughput or less congestion. Thus, the highway for instance can provide very low congestion, while mixed traffic provides more throughput but higher congestion, which entails higher risks. In my opinion, the two-layered can be a very good trade-off between the three options.

   - ***Can the model show if the MRS is robust?***

You have made disruption scenarios where the system demonstrates its robustness even when some robots fail. However, robustness has multiple dimensions, of which robot failure is only one. Other robustness dimensions are energy consumption, failure of pick-up or drop-off stations, and even the ability of the system to cope with fluctuation of demand, especially with extremely high and low peaks. These could also be added to you project to further show the robustness of the system.

   - ***Can the model show if the MRS is flexible?***

Yes, this is definitely possible with your simulation, given that robots can handle different types of packages and can be easily expandable and adaptable to other layouts.

- ***Can the model show if the MRS is performing well?***

The simulation can show how many packages a system is able to sort in one shift, using the throughput indicator. However, SLA and distance travelled are also important indicators for performance that should be considered, especially in your case the distance travelled. All the other KPIs can be used to prove the good performance of the system.

3. ***What other results/experimental designs do you think should be developed in order to fully answer the above-mentioned questions?***

I would suggest you to focus a bit more on disruption. Equipment failure, arrival stations failure, demand fluctuations etcetera. These are all possible to show in your simulation. It would be nice to see the time impact of disruption. Beyond this, I think that all the scenarios you have made are very suitable for the project.

4. ***Any other remarks/considerations you want to make or questions you want to ask?***

Another remark is that I would explain why the highway has been implemented in that way and not for example with a path in the center of the transport area. In that case there would be less travelled distance. Evidently, in that case the disadvantage would be again the crossing between unloaded and loaded robots, but you can explain why that configuration. For instance, I did an AGV project and implemented the area as a circle. The distance travelled was very long and everyone thought it was a stupid choice. However, the reason was that we made a trade off between distance travelled and safety and predictability of the system. That configuration offered us higher level of safety and higher predictability. Therefore, we decided to have less performance but higher safety and predictability. You can explain why this configuration and make recommendations regarding the use of other configurations for future studies.

Question: Have you calculated the number of interferences among robots?
Answer: the congestion indicator has been calculated in a way that it shows the amount of interferences among robots. In fact, congestion is calculated measuring the speed of every robot and divide it by the max speed of robots. So, the higher the congestion, the higher the number of interactions between robots. Furthermore, we implemented a heat-map that showed where most of the robots concentrate in the space. Initially, most of the congestion is in the pick-up areas, but after a warm-up period, less congestion is observed also in that area.

Question: Are you taking into account data during the warm-up period?
Answer: no, this data is removed from the statistics. We have seen that for some measurements the warm-up period can arrive until 1000 ticks, so we have decided to collect all data after 1000 ticks.

Overall, I think you did a lot of work and with great results. I would make sure that travelled distance of robots and parcels is a KPI for your system, draw graphs, and explain what you see in graphs. Help the decision-maker to make savvy decisions!

Appendix L

The workspace required by robots can be calculated by modelling robots as circles (see figure 54). We know that robots have a dimension of 90x60 cm (length x width). In addition, we want to leave at least 50 cm extra space to the robots to move. Therefore, we can model robots as circles having a radius of 95 cm (90/2 = 45 + 50 = 95 cm). Hence, to calculate the space we consider circles of 190 cm of diameter.

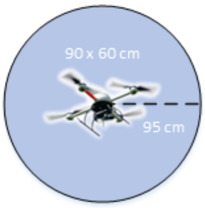
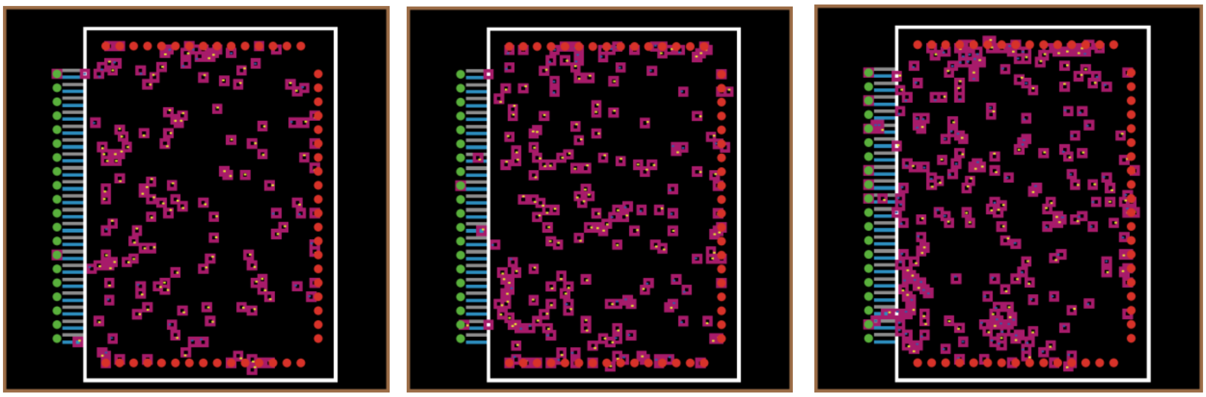


Figure 54. Robots modelled as circles

In mathematics, this problem corresponds to a circle packing problem, where the objective is to pack n units into the smallest possible square. To calculate the minimum area required by robots, we have used WolframAlpha and refer to E. Specht (2010). Using the Hexagonal packing, the workspace required by 100 robots is 361 m^2; the workspace required by 150 robots is 519.84 m^2; and, the workspace required by 200 robots is 706.5 m^2. This is a static result that does not consider the space needed by robots to move freely. Further, this space does not consider the space required by pick-up and drop-off buffers. Figures 55a-55b-55c show the area covered by 150, 200 and 250 robots, which have a security distance (in pink). As can be observed, within this area (3750 m^2), 150 robots have enough free space (in black) to move in the environment (figure 54a). Moreover, robots are well distributed in the area, meaning that aggregates of robots are less visible that constrain the action space of robots.



(a) (b) (c)

Figures 55a-55b-55c. Coverage of workspace by 150 (a) – 200 (b) – 250 (c) robots

With 200 robots, we can notice that clusters of robots are becoming more evident, with robots having in some cases, lower space for movement (figure 55b). These aggregates of robots in confined space can lead to deadlocks situations, as seen with 250 robots (figure 55c). In a scenario with 250 robots, robots do not have enough free space, bringing about circular deadlocks. These static results also offer a good overview of the congestion in the area, measured in the experimentation phase.

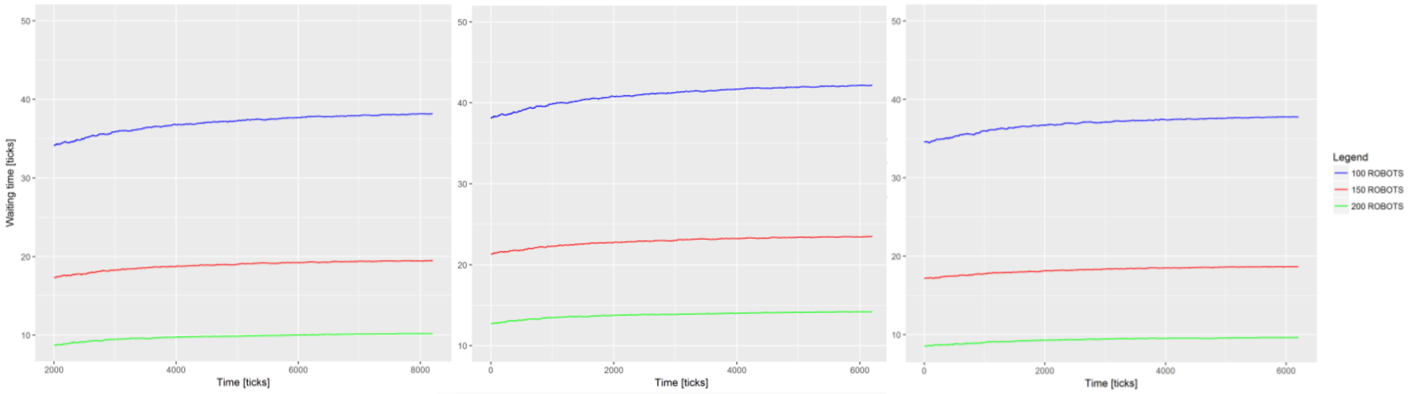
Appendix M

Other results experimental designs 100% light and low volume parcels

In this Appendix, we display and describe other results obtained in the experimental design with 100% light and low volume parcels in the different traffic configurations. These results support the conclusions made in the main report.

M.1 Waiting time in Mixed Traffic – Highway – Two-Layered

In this scenario, the waiting time, i.e. time waited by parcels at pick-up buffers before being collected by robots, reduces with the increase of the number of robots. This result was easy to predict, considering that more robots are available to pick-and-transport parcels. Interestingly, the waiting time reduces by a large degree when the number of robots is increased from 100 to 150, while a small reduction is observed with 200 robots. This implies that in this area, 100 robots are not enough to perform quickly the tasks. Another clear observation that we can make is that in the Highway the waiting time is higher in comparison to the other two traffic configurations. In Mixed Traffic and Two-layered, the waiting time is approximately the same, meaning that parcels wait roughly the same time at the stations before being collected by robots. This result corroborates the throughput outcome seen in the main report in this scenario. In fact, looking at the throughput and waiting time, we can observe that in this scenario, there is not a significant difference between Mixed Traffic and Two-layered, while a wider gap exists with Highway.



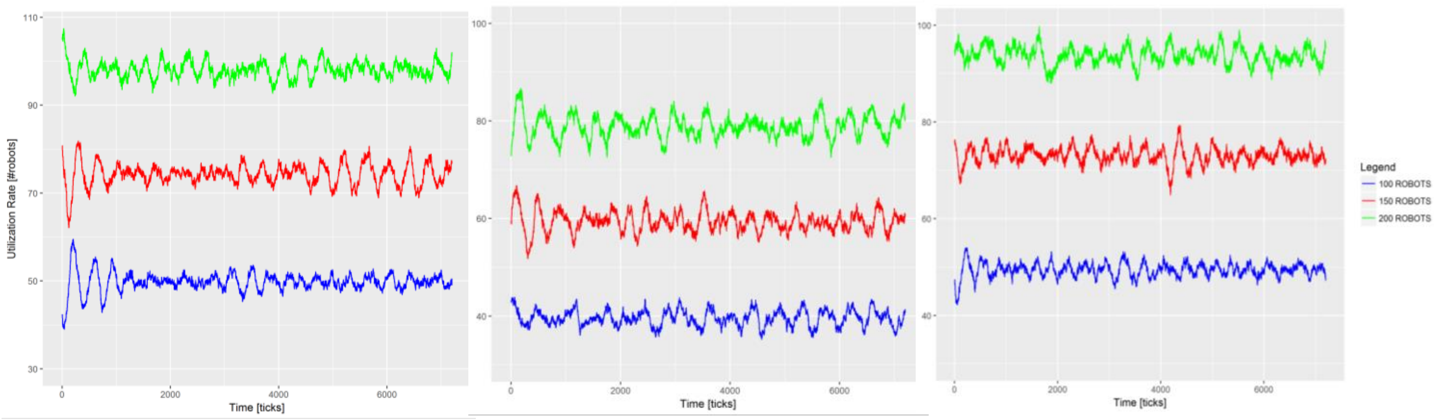
(a) (b) (c)

Figures 56a-56b-56c. Waiting time in Mixed Traffic (a), Highway (b), Two-layered (c) with 100% light and low volume parcels and 100-150-200 robots

Furthermore, this result confirms our hypothesis on the relation between congestion and throughput. In fact, although the congestion is higher in Mixed Traffic than in Two-layered, the waiting time of parcels is not very different. This implies that within this scenario, congestion does not have a significant impact on throughput.

M.2 Utilization Rate in Mixed Traffic – Highway – Two-Layered

The utilization rate is another performance indicator that indicates the average number of loaded robots, i.e. robots with parcels. This indicator gives insights into the amount of idleness in the system. Figures 56a-56b-56c feature the results obtained in a scenario with 100% small parcels in the different traffic configurations. As can be viewed, after 1000 ticks we cannot still have steady results, especially in Mixed Traffic where results fluctuate vigorously. Steady results are obtained efficiently in Two-layered. From these results, we can notice that the Mixed traffic and Two-layered present again a similar outcome; whereas, in the Highway, the amount of idleness is higher. This inference corroborates the results described in the main report on distance travelled idle. In fact, in Mixed Traffic and Two-layered we have on average almost half robots travelling with parcels (loaded) and half travelling without parcels (unloaded). In highway, instead, the majority of robots travel without parcels since the travelling distances are longer than in Mixed Traffic and Two-layered. Therefore, these results reinforce the conclusions made in the main report.



(a) (b) (c)

Figures 57a-57b-57c. Utilization rate in Mixed Traffic (a), Highway (b), Two-layered (c) with 100% light and low volume parcels and 100-150-200 robots

Appendix N

Results experimental designs 95% light and low volume parcels

In this Appendix, we display and describe other results obtained in the experimental design with 95% light and low volume parcels in the different traffic configurations. As earlier argued, the impact of cooperative transport becomes already visible with 5% heavy and high volume parcels. However, in the main report, we have not reported these results, as this impact becomes more evident with 10% heavy and high volume parcels. Nevertheless, the results shown in this appendix further confirms the conclusions inferred in the main report.

N.1 Throughput in Mixed Traffic – Highway – Two-Layered

Throughput is one of the most important performance indicator for parcel operators, indicating the number of parcels delivered per unit of time. In comparison to the results obtained in the scenarios with 100% light and low volume parcels, the number of parcels delivered in half hour decreases strongly in this scenario. In this scenario, more resources are needed to transport 5% of the parcels. In two-layered (max throughput), the throughput decreases by 25% with 100 robots, 22.6% with 150 robots and 15.7% with 200 robots. In the other two configurations, the throughput also decreases less with increasing number of robots. This confirms our conclusion that cooperative transport requires more robots into the field to be effective.

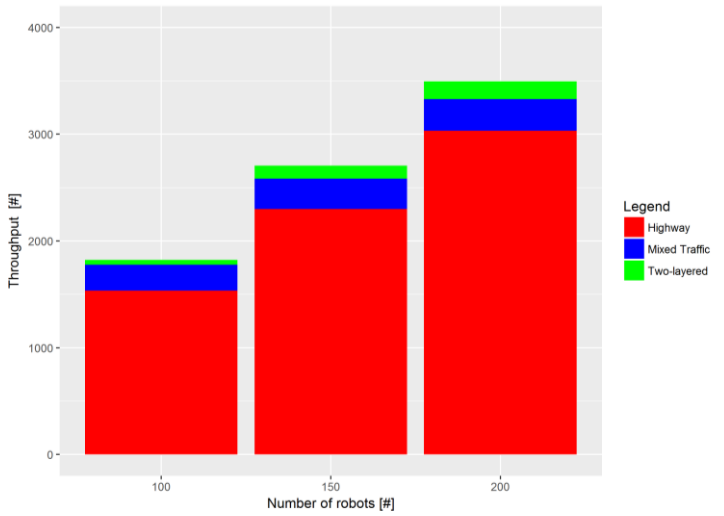
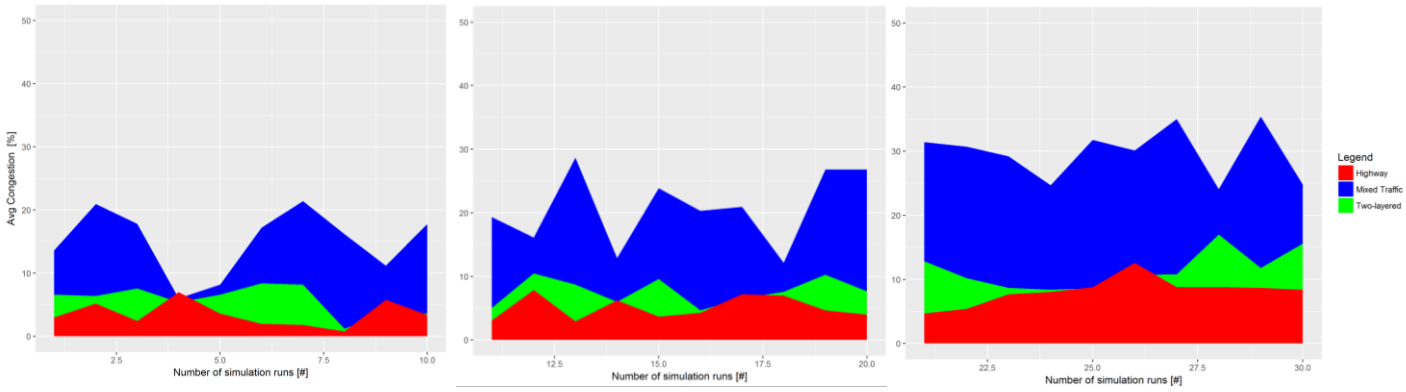


Figure 58. Throughput in the different traffic configurations with 95% heavy and high volume parcels and 100-150-200 robots

From these results, it is also evident that Highway offers lower throughput than Mixed traffic and Two-layered. In addition, the gap in throughput between Mixed traffic and Two-layered becomes more evident in this scenario, particularly with 150 and 200 robots.

N.2 Congestion in Mixed Traffic – Highway – Two-Layered

Congestion is measured as a function of the speed of robots. This is computed as the ratio between the average speed of robots and their maximum speed; this quantity is divided by the number of robots. In the main report, looking at the results from the scenarios with 100% and 90% light and low volume parcels, we have already inferred that in Mixed Traffic the congestion is higher than in Highway and Two-layered. These results corroborate this conclusion. In addition, it is apparent how congestion increases with the number of robots, particularly in Mixed Traffic. Clearly, this configuration does not withstand well a high number of robots. Highway presents the lowest level of congestion, thus making this configuration the safest traffic alternative. Congestion is in fact a good representative of the level of safety in the transport area, given that it provides insights into the amount of interferences among robots in the various traffic configurations. Therefore, it can be concluded that Highway and Two-layered provides more safety than Mixed traffic, where the motion of the different robot entities is not separated. Furthermore, comparing the results obtained in this scenario with the results obtained in a scenario with 100% light and low volume parcels, it is evident that cooperative transport reduces the level of congestion. The lower density of robots in the space causes this reduction. Indeed, in this scenario, formations of robots are built to transport big parcels, forming large entities that move with short distances occupying lower space compared to single entities dispersed into the environment. Therefore, these results also reinforce the conclusions inferred previously about the lower level of congestion provided by cooperative robots.

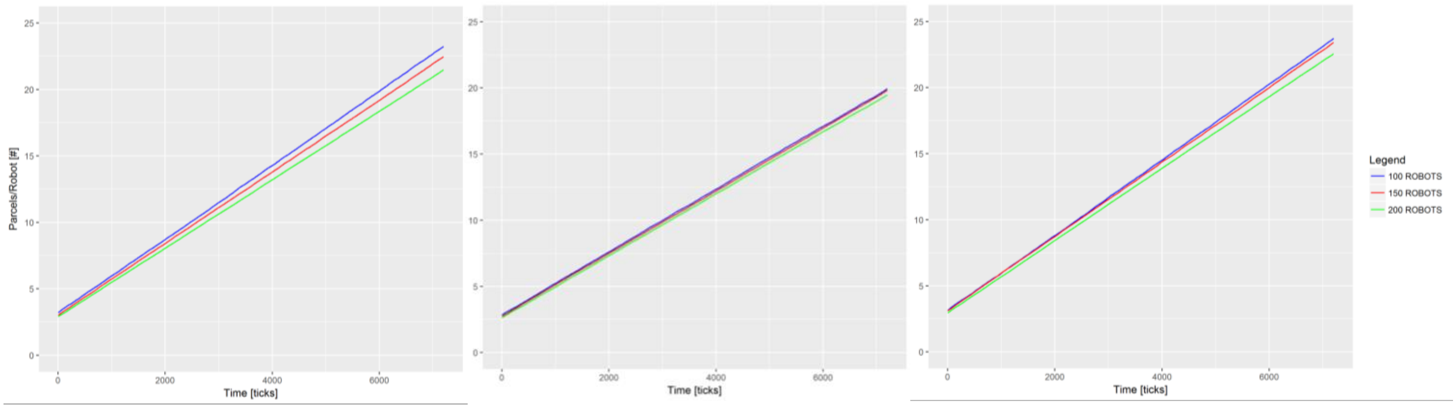


(a) (b) (c)

Figures 59a-59b-59c. Congestion in Mixed Traffic, Highway, Two-layered with 95% light and low volume parcels and 100 (a), 150 (b) and 200 (c) robots

N.3 Performance in Mixed Traffic – Highway – Two-Layered

Performance indicates the number of parcels transported per unit of time by each robot. This indicator can be easily computed by dividing throughput by the number of robots. As can be observed in Figure 60a, in Mixed traffic the performance of robots reduces with the increase of robots. In particular, in this traffic configuration the performance decreases strongly after 150 robots, implying that this configuration does not tolerate well a number of robots larger than 150. Interestingly, in Highway (Figure 60b) the lines of performance are almost identical, with a slight difference shown in the line with 200 robots. This implies that this traffic configuration tolerates sufficiently high number of robots. Further, this entails that adding more robots in the transport field guarantees an almost linear increase of throughput. In Two-layered (Figure 60c), performance is almost the same with 100 and 150 robots, with nearly coincident lines. However, the performance of robots drop when the number of robots is larger than 150 robots. When observing these results it seems that Highway withstands higher number of robots better than Mixed Traffic and Two-layered do. Moreover, as already inferred in the conclusions earlier, cooperative transport requires higher number of robots to be effective. This is already visible in this scenario, with the performance lines of 150 and 200 robots increasing in comparison to the scenario with 100% light and low volume parcels. This result is even more evident in a scenario with 90% light and low volume parcels.

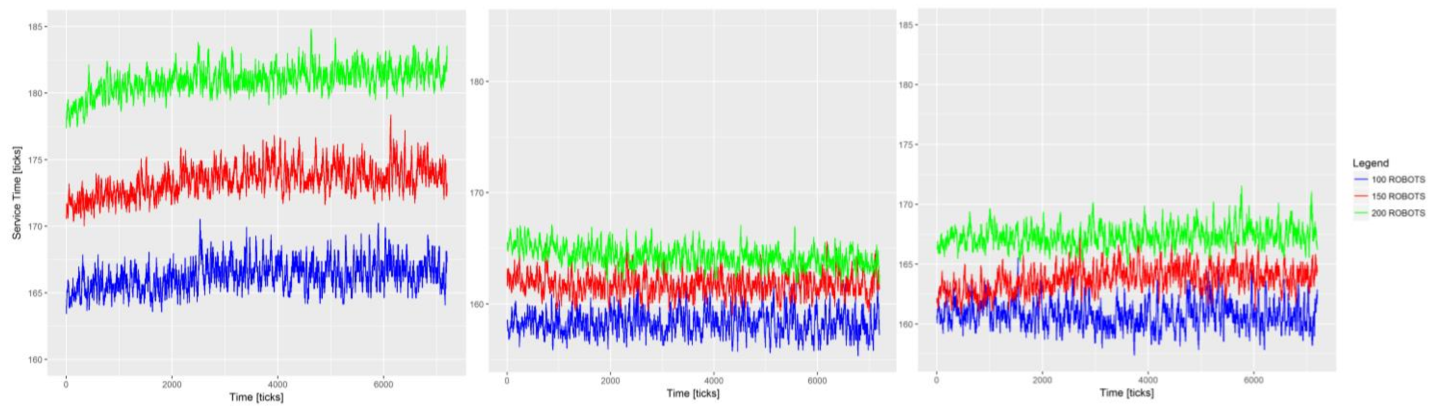


(a) (b) (c)

Figures 60a-60b-60c. Performance in Mixed Traffic (a), Highway (b), Two-layered (c) with 95% light and low volume parcels and 100, 150 and 200 robots

N.4 Service Time light-low volume parcels in Mixed Traffic – Highway – Two-Layered

Service time refers to the time robots take to transport a parcel from pick-up to drop-off buffers. As can be observed in Figures 61a-61b-61c, the results of service time for light and low volume parcels do not change in this scenario in comparison to the results from the scenario with 100% light and low volume parcels. This implies that cooperative transport does not influence the service time of light and low volume parcels.



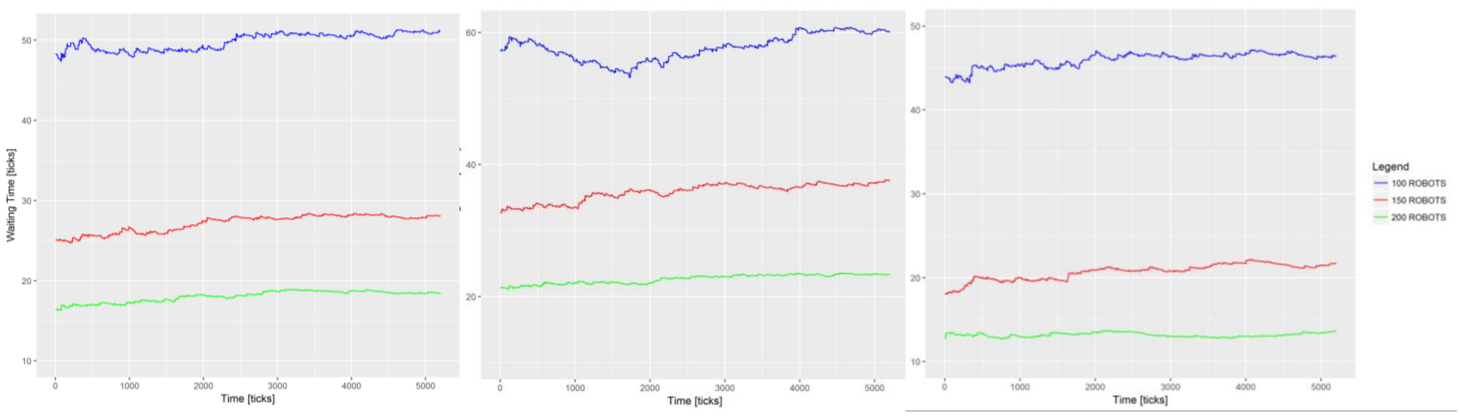
(a) (b) (c)

Figures 61a-61b-61c. Service time in Mixed Traffic (a), Highway (b), Two-layered (c) with 95% light and low volume parcels and 100, 150 and 200 robots

Moreover, from these results, it is evident how the service time is extensively higher in Mixed Traffic, where the traffic flow is not separated according to the loaded/unloaded state of robots. While there is not a significant difference between Highway and Two-layered, where the motion of loaded robots is separated from the motion of unloaded robots. In addition, in all configurations, the service time rises with the increase of the number of robots. These results, therefore, confirms what we have already inferred in the main report.

N.5 Waiting Time heavy-high volume parcels in Mixed Traffic – Highway – Two-Layered

Waiting time is evidently higher for heavy and high volume parcels, given that four robots are required to transport these parcels. This is apparent when looking at Figures 62a-62b-62c, featuring the waiting time of big parcels. The waiting time of heavy and high volume parcels is roughly 25% higher than the waiting time of light and low volume parcels. Furthermore, it is also evident how in Highway, these parcels wait longer before being collected, in comparison to Mixed traffic and Two-layered. This is due to the fact that in Highway robots travel longer distances before arriving at the pick-up points. It is also interesting to notice that the difference in waiting time between Mixed traffic and Highway reduces with increasing number of robots. Whereas, the difference in waiting time between Mixed traffic and Two-layered widens with increasing number of robots. This confirms that Mixed traffic tolerates poorly high number of robots, compared to the other two configurations. This entails that with high number of robots, and assuming the area stays the same, Highway and Two-layered should be preferred over Mixed traffic.



(a) (b) (c)

Figures 62a-62b-62c. Waiting time of heavy and high volume parcels in Mixed Traffic (a), Highway (b), Two-layered (c) with 95% light and low volume parcels and 100, 150 and 200 robots

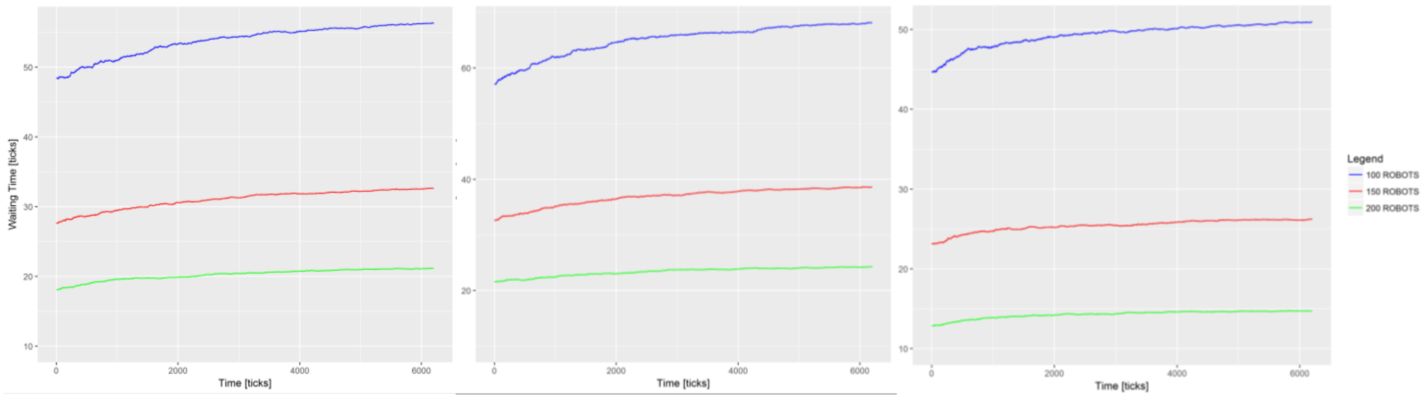
Appendix O

Other results from experimental designs 90% light and low volume parcels

In this Appendix, we display and describe other results obtained in the experimental design with 90% light and low volume parcels in the different traffic configurations. Many results from these scenarios have been shown in the report. The results in this Appendix aim to confirm what was inferred earlier.

O.1 Waiting Time in Mixed Traffic – Highway – Two-Layered

In Chapter 5, we have inferred the conclusion that congestion does not influence throughput in a scenario with only light and low volume parcels, but that it does influence throughput when the percentage of heavy and high volume parcels rises. This is due to the fact that in these scenarios, buffers require more robots. The results on waiting time, shown in Figures 63a-63b-63c, validate this hypothesis.

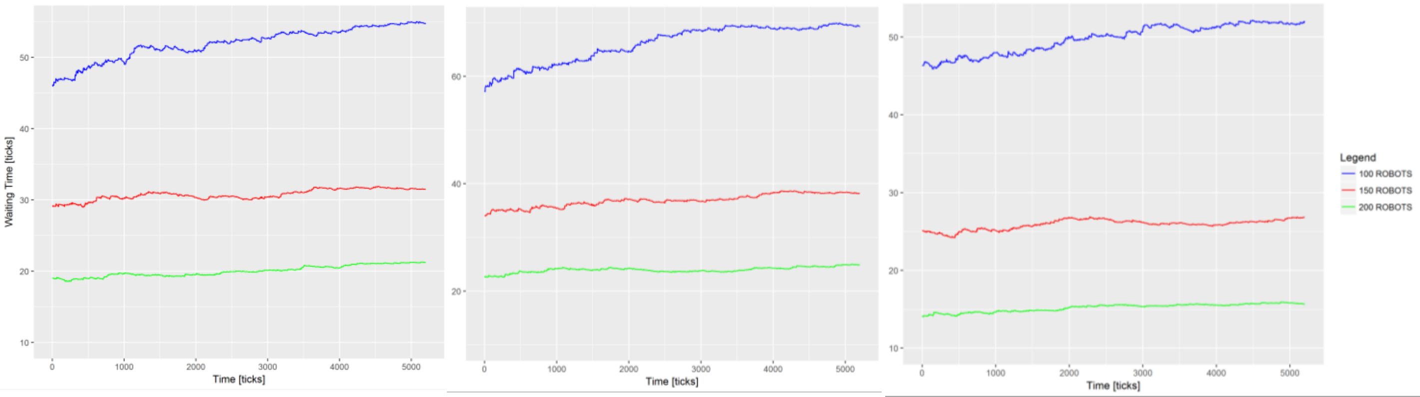


(a) (b) (c)

Figures 63a-63b-63c. Waiting time of light and low volume parcels in Mixed Traffic (a), Highway (b), Two-layered (c) with 90% light and low volume parcels and 100, 150 and 200 robots

In fact, when comparing the waiting time in Mixed traffic (63a) and in Two-layered (63c), we can notice how the difference in waiting time is noticeable. This was not visible instead in the results on waiting time from the scenarios with 100% light and low volume parcels. This entails that high level of congestion in this scenario leads to an increase in waiting time of parcels. Therefore, this supports our conclusion on the relation between congestion and throughput.

The same conclusions can be inferred when looking at the waiting time of heavy and high volume parcels (Figures 64a-64b-64c). The difference between waiting time in Mixed traffic and Two-layered manifests in these figures. Further, the waiting time remains highest in Highway, due to the longer distances travelled. However, for high number of robots, the difference in waiting time between Mixed traffic and Highway reduces. This also confirms what we have argued earlier, which is to say, Mixed traffic reduces its effectiveness with increasing number of robots. A wider difference in waiting time exists between Highway and Two-layered, which configuration also supports sufficiently high number of robots.

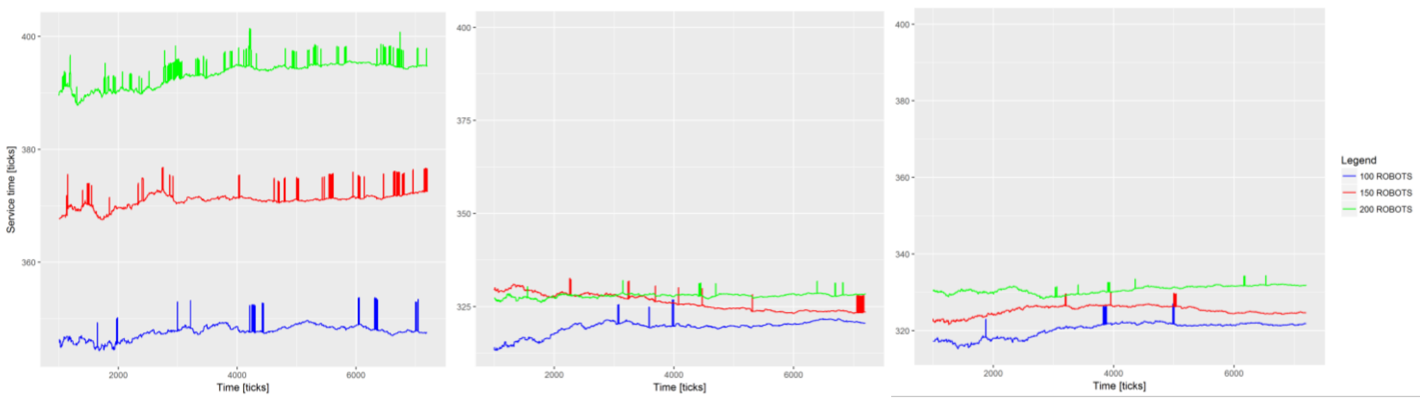


(a) (b) (c)

Figures 64a-64b-64c. Waiting time of heavy and high volume parcels in Mixed Traffic (a), Highway (b), Two-layered (c) with 90% light and low volume parcels and 100, 150 and 200 robots

O.2 Service time heavy-high volume parcels in Mixed Traffic – Highway – Two-Layered

The service time for heavy and high volume parcels is certainly higher than the service time for light and low volume parcels, since robots transporting these types of parcels travel at half the speed of robots travelling with small parcels. This result is apparent when observing Figures 65a-65b-65c. In Mixed traffic, where service time is higher due to the high level of congestion, robots take up to 90s to transport a big parcels, in a scenario with 200 robots. In the same scenario (with 200 robots) and in Highway, robots take around 81.25s to transport these parcels. Approximately the same results are obtained in Highway and in Two-layered. The difference in service time between the different traffic alternatives becomes more conspicuous with the increase of robots. In fact, with 100 robots the three traffic scenarios present almost the same results, while with 150 and 200 robots the service time increases enormously in Mixed Traffic, whereas it remains nearly stable in the other two traffic configurations.



(a) (b) (c)

Figures 65a-65b-65c. Service time for heavy and high volume parcels in Mixed Traffic (a), Highway (b), Two-layered (c) with 90% light and low volume parcels and 100, 150 and 200 robots

Appendix P

Results experimental designs 85% light and low volume parcels

In this Appendix, we report and discuss the results obtained in the experimental design with 85% light and low volume parcels in the different traffic configurations. The impact of cooperative transport becomes even more evident in this scenario, considering the high percentage of heavy and high volume parcels. The results in this appendix further confirms the validity of the conclusions inferred in the main report.

P.1 Throughput in Mixed Traffic – Highway – Two-Layered

As viewed from the scenarios with 95% and 90% light and low volume parcels, throughput decreases when the increase of percentage of heavy and high volume parcels. This becomes more evident when observing the results obtained from the experimental design with 85% light and low volume parcels (see Figure 66). In comparison to the results obtained in the experimental design with 100% light and low volume parcels, the throughput decreases remarkably in this scenario. In fact, in Two-layered, where the throughput reaches its highest values, the throughput reduces by 91.67% with 100 robots, 71% with 150 robots and 62% with 200 robots in comparison to the results obtained with 100% small parcels.

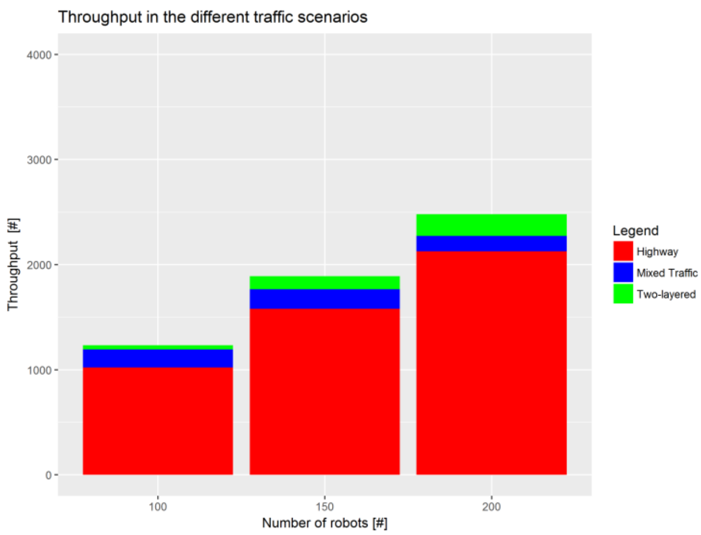


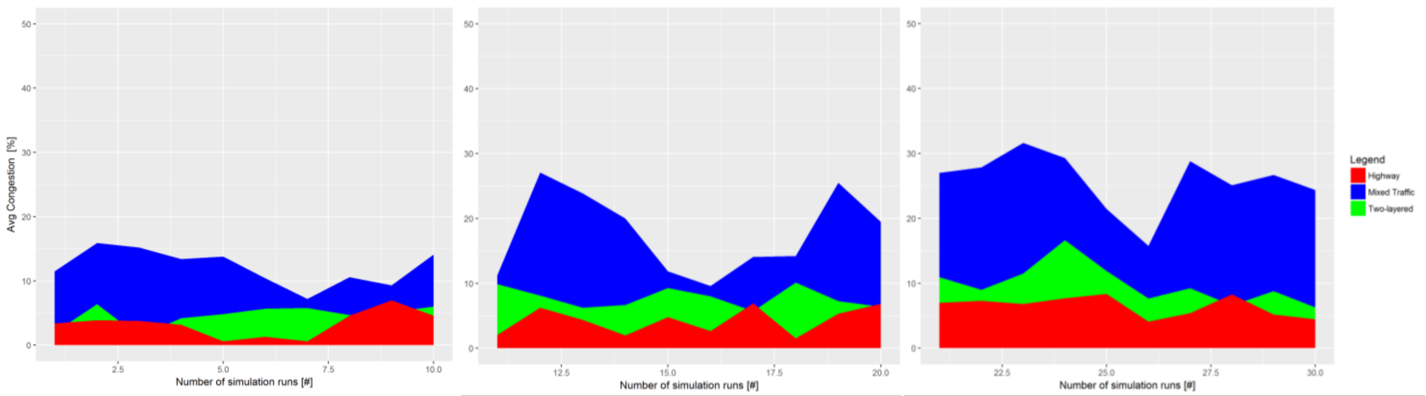
Figure 66. Throughput in Mixed Traffic, Highway, Two-layered with 85% light and low volume parcels and 100, 150 and 200 robots

This result confirms what we have inferred from the results with 90% light and low volume parcels. Cooperative transport negatively and strongly affects throughput. Further, cooperative transport requires higher number of robots to be effective. Interestingly, from these results, we can notice how the difference in throughput between Mixed Traffic and Two-layered widens even more in this scenario, with Two-layered producing superior results. In addition, the difference in throughput between

Mixed traffic and Highway shrinks even more, especially with high number of robots. This entails that in this scenarios, these two traffic configurations offer very similar results. At a certain point, when the percentage of light and low volume parcels reduces beyond 85%, we can expect that Highway will outpace Mixed traffic in terms of system effectiveness. A wider different still remains between Highway and Two-layered, with the latter showing off again its superiority in comparison to the other traffic configurations. These results further corroborate the conclusions inferred in the main report.

P.2 Congestion in Mixed Traffic – Highway – Two-Layered

Figures 67a-67b-67c provide an overview of the level of congestion in the different traffic alternatives in a scenario with 85% light and low volume parcels and with 100, 150 and 200 robots. In comparison to the results from the other experimental designs (100%, 95% and 90% small parcels), in this scenario the level of congestion decreases further. This highlights the fact that cooperative transport reduces the level of congestion, in view of the lower distribution of robots in the space. Therefore, these results supports the inferences made in the report. Moreover, the level of congestion is always the highest in Mixed traffic, while a small difference exists between Highway and Two-layered. These results were therefore predictable and desirable, as they confirm our conclusions.

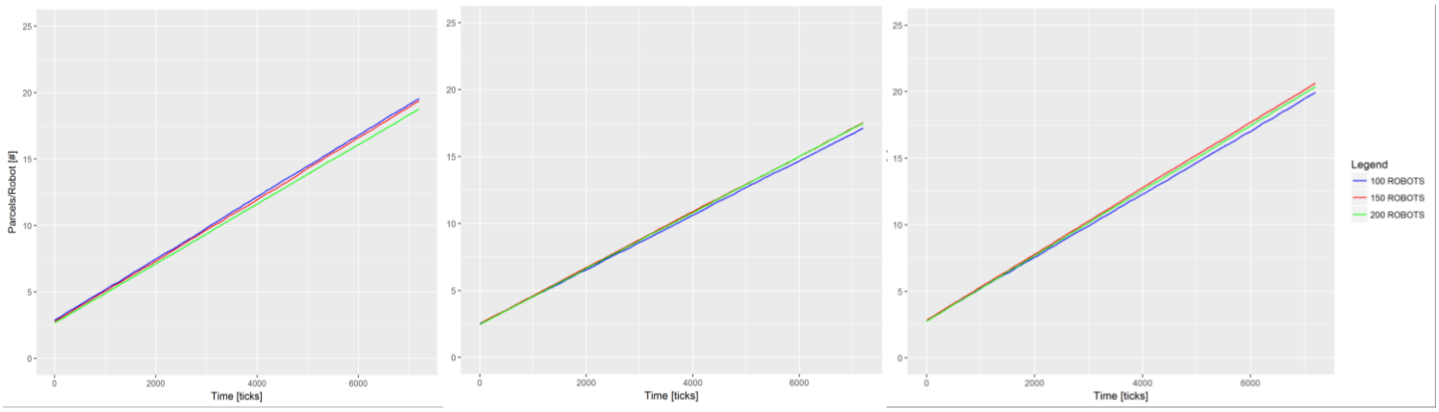


(a) (b) (c)

Figures 67a-67b-67c. Congestion in Mixed Traffic, Highway, Two-layered with 85% light and low volume parcels and 100 (a), 150 (b) and 200 (c) robots

P.3 Performance in Mixed Traffic – Highway – Two-Layered

Figures 68a-68b-68c features the results on robot performance (i.e. parcels/robot), from this experimental design with 85% light and low volume parcels. Interestingly, in these figures we can notice how the performance of robots in this scenario increases additionally with the increase of the number of robots. In Mixed Traffic, the lines of 100 and 150 parcels converge, while a small gap still remains with the line of 200 robots. In Highway, the lines of 150 and 200 robots overstep the line with 100 robots, implying that in this scenario the higher the number of robots the higher the parcels collected by single robots. This entails that in this scenario and in Highway, 150 and 200 robots should always be preferred over 100 robots. Similarly, in Two-layered, the performance lines of 150 and 200 robots exceed the line with 100 robots. This again suggests that in this traffic configuration and with 85% small parcels, more robots should be introduced. These results validate even further the conclusions reported earlier. First, it becomes clear that Mixed Traffic withstands poorly over 150 robots. Second, it becomes even more apparent how high percentage of big parcels necessitates high number of robots to produce good results. Therefore, these results make our inferences more convincing.

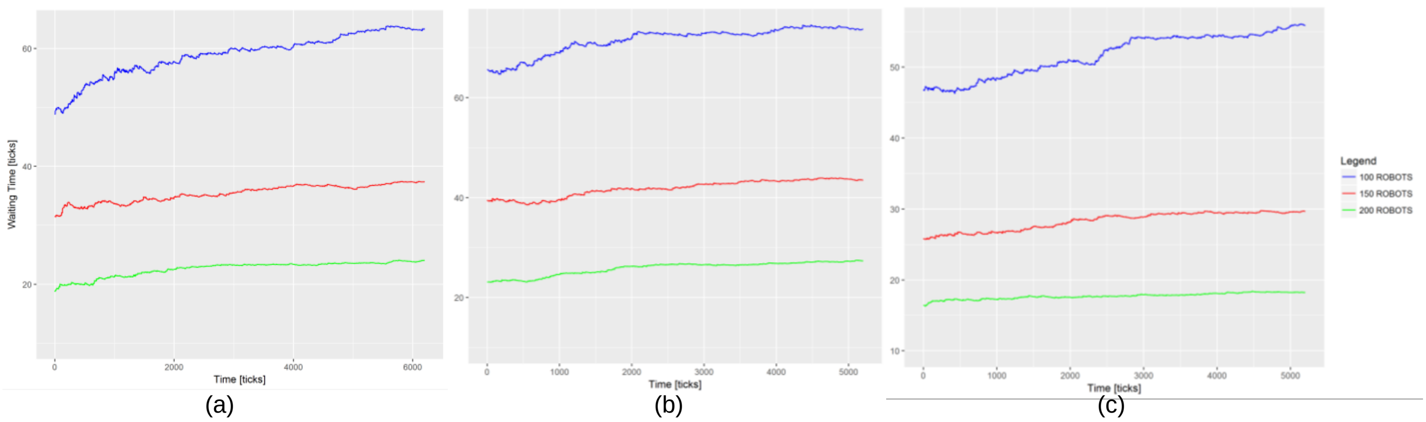


(a) (b) (c)

Figures 68a-68b-68c. Robot performance in Mixed Traffic, Highway, Two-layered with 85% light and low volume parcels and 100 (a), 150 (b) and 200 (c) robots

P.4 Waiting Time in Mixed Traffic – Highway – Two-Layered

Figures 69a-69b-69c exhibit the waiting time of parcels in this experimental design with 85% light and low volume parcels. From these results, we can observe that the difference between the waiting time in Mixed traffic and in Two-layered has increased, due to the higher congestion brought along by the former traffic configuration. Moreover, the difference in waiting time between Mixed traffic and Highway reduces, showing that Mixed traffic is more exposed towards the negative effects of cooperative transport. These results strengthen our conclusions, further proving the negative impact of congestion on system effectiveness with higher percentage of heavy and high volume parcels. Moreover, these results show that Two-layered offers superior effectiveness in comparison to Mixed Traffic and Highway, while Highway nearly achieves the same effectiveness provided by Mixed traffic while offering higher safety.



(a) (b) (c)

Figures 69a-69b-69c. Waiting time in Mixed Traffic, Highway, Two-layered with 85% light and low volume parcels and 100 (a), 150 (b) and 200 (c) robots