Exploring Local Communication to Improve Swarm Box Pushing

By

Cao, Cong

MSc Robotics Dissertation





Department of Engineering Mathematics
UNIVERSITY OF BRISTOL
&
Department of Engineering Design and Mathematics
UNIVERSITY OF THE WEST OF ENGLAND

A MSc dissertation submitted to the University of Bristol and the University of the West of England in accordance with the requirements of the degree of MASTER OF SCIENCE IN ROBOTICS in the Faculty of Engineering.

Declaration of own work

I declare that the work in this MSc dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

Cong Cao September 12, 2022

Acknowledgement

First and foremost, I would like to dedicate my deepest thanks to Dr. Paul O'Dowd. His careful and attentive guidance persisted throughout the whole of my dissertation. When I felt powerless and disoriented, his support and encouragement was a beacon of light that helped me find my way. I have benefited from his academic rigour and comprehensive thinking, which has provided me with deeper insights in my research. Without his illuminating instruction, the dissertation could not have reached its current state.

Also I would like to thank my family for their support all the way through my study. Although we are separated by a great distance physically, their encouragement made me feel instantly cared for and encouraged.

In the end, I would like to thank all my friends, without them I would not be able to get through my tough time. Their company is the best comfort for me.

Abstract

Swarm robotics systems that are inspired by herd animals are implemented on a number of simpler and less capable robots so that groups of robots could do tasks that are tough for individual robots to accomplish. The benefits of swarm robots include robustness, scalability, adaptability, economy, and the capacity to do multiple tasks in parallel. Enabling cooperation amongst robots is vital for enhancing the capabilities of swarm robotics, but it is also challenging. Communications in cooperation among collective animals are indispensable, and this project investigates whether and which local communication method might improve the efficiency of robots doing box-pushing tasks in groups. I developed systems with implicit and explicit communication, as well as a system with no communication for comparison. The system that introduced implicit communication used a box-pushing strategy with four different phases: exploration phase, object recognition phase, and box-pushing phase. In system with explicit communication, a recruiting phase is added to promote cooperative behaviors. The project's findings indicate that implicit communication has better benefits, and explicit communication has acquired considerable gains after improvement have been made shows that explicit communication has larger room for enhancement.

Number of words in the dissertation: 15710 words.

Contents

Page

1 Introduction **6**

1.1 Literature Review 7

1.2 Summary & Contributions 17

2 Research Methodology **23**

2.1 The Box-Pushing Task 23

2.2 System Overview 24

2.3 Implementation 29

2.4 Experiment Methodology 42

3 Results **47**

3.1 Varying Communication types 47

3.2 Varying Number of Robots 50

3.3 Summary 52

4 Follow-up Experiments **54**

4.1 Additional experiment 1: Single robot box pushing experiment 54

4.2 Additional experiment 2: Using explicit communication-recruitment with 10s waiting time 57

4.3 Additional experiment 3: Message sending and receiving test 59

4.4 Summary 61

5 Conclusion and future work **63**

5.1 Conclusion 63

5.2 Future work 66

A Appendix **68**

References **73**

1 Introduction

This project is based on the realistic context of moving objects in a warehouse and the desire to find an inexpensive solution for robots. So the task scenario I am going to set is that of multiple robots cooperating to push boxes to a target location. This scenario is intended to explore and evaluate the impact of local communication between the co-operating robots. So this study investigates various communication strategies, including no communication as baseline.

In recent years, research on swarm robots has flourished[1]. Swarm robotics systems are inspired by social animals who exhibit collective behaviours. Collective behaviours provide an alternative to conventional robotics, such as where a task is solved by one robot alone. Advancing swarm robotics to complete more sophisticated tasks would be beneficial to the industrialization of human society[2].

Swarm Robotics research is motivated by ant colony, bees, flock, etc[3]. The collective behavior of these animals does not need coordination by a central authority since it consists of the superposition of individual actions to generate a complex collective behavior. Such a group has the beneficial characteristics of scalability, resilience, and adaptability, and it can handle several subtasks in concurrently[4]. The transfer of such characteristics from animal populations to robotic swarm system can improve the system's capabilities in several aspects. These concepts and their relevance to this study are discussed further in section 1.1.

With the rapid development of technology, there is an increase in the number of applications fields for robots, including tasks such as security patrols, hazardous environment rescue, warehouse automation, chemical waste sorting, etc[5]. These tasks require the cooperation of a large number of robots due to the amount of work involved, the research of swarm collaboration will be described in the literature review section 1.1. And in this project, I will choose to start with such cooperative tasks and conduct in-depth research on the application of local communication in improving cooperation of swarm robots.

In chapter 1, the background of swarm robotics and some works in the applications in swarm

have been introduced. I have proposed my research questions, documented my hypotheses and the my expectation of the experiments. These hypotheses are the foundation of the experiments conducted and their design.

In chapter 2, the concept and implementation of the system for box pushing task using multiple communication methods are provided. The layout of the experimental arena, the variables to be measured and the procedure of the experiment is also shown in this chapter.

In chapter 3, the result of the main experiment is demonstrated and discussed. And some issues that need further discussion are raised

In chapter 4, three sets of follow-up experiments are designed and carried out in response to the questions that needed further discussion. The results of the experiments are presented and discussed.

In chapter 5, all the conclusions obtained from the results are summarized and discussed, and the areas for improvement of the project are prospected.

1.1 Literature Review

In this section, I will explain the background and definition of swarm robotics in detail by presenting some fundamental ideas and theories. Swarm robotic is a specific area of robotics as a whole, which has some design constraints to facilitate the underlying mechanisms of operation. Because this study is within swarm robotics, these constraints must be discussed.

Next I will mention various related investigations of swarm robots concerning collaboration, focusing on the communication form of robots in collaboration and the study on the cooperation of robots pushing boxes, the box-pushing task will be described in section 2.1. This discussion serves to help locate the box-pushing task selected for this study.

Finally, I will highlight research gap based on past works and put forward my research. This final summary includes my experiment hypotheses (see section 1.2.3). These hypotheses are presented after the literature review to ensure they can be understood with a high level of detail, before moving on to the Research Methodology chapter (chapter 2).

1.1.1 Swarm Robotics

Definition of swarm robotics

About 20 years ago, swarm robots were named "cellular robots" as they were a generic type of cellular automata, representing a group of simple robots that can organize themselves[6]. Currently, *Swarm Robotics* has progressed to formulate a concise and comprehensive definition. Taking cues from social insects and animals, swarm robotics investigates how to construct a population of entities comprising a plurality of basic robots, so that the robots in this population can accomplish particular collective behavior from the engagement between several agents and between agents and their environment[3].

The above is the definition of Swarm Robotics in general. To separate Swarm Robotics from other multi-robot studies, there are constraints that restrict the range of Swarm robotics that are absent from multi-robot systems. In addition, a central authority which coordinates the behavior of each robot in the research area of *multi-robot systems*, whilst Swarm Robotics is defined by self-organization. The following are the key characteristics relevant to swarm robotics:

- Decentralization: Swarm robotic systems are not intended to have an internal or external central authority that can organize or instruct robots. Rather, the robots must interact with other robots or the environment and choose their group behavior collaboratively, this ability of determining one's own group behavior is referred to as *Self-organisation*.

- Autonomy: For a robot system to be considered a Swarm robotic system, its constituent robots must be able to detect their surroundings via sensors and engage in real-world physical contact.

- Scale: It is unrealistic to establish a lower limit for the number of robots in a swarm robotic system, since this would require analysis based on various applications. Therefore, the expectation of a swarm robotic system is that the number of robots is adjustable for research purposes, but the system should continue to function properly when the number of robots is expanded to fulfill the needs of the circumstance.

- Homogeneity: Every robot in swarm robotic system should be homogeneous. They are

equivalent in function, structure, and status. No authority exists to guide them. Through contact with the environment, robots may adopt diverse behaviors, but this does not imply that their roles are distinct.

- Incapability: As an alternative to single robot or multi-robot systems, Swarm Robotics has the benefit of providing a low-cost solution that enables the employment of a large number of robots without the requirement for advanced technology on a single robot. In tasks that are challenging for a single robot, a large number of low-value, incompetent swarm robots may nonetheless deliver superior outcomes compared to a single robot working alone.

- Local sensing and communication: Individual robots in a swarm should only collect information from the environment (local sensing) or by communicating with their neighbors (local communication) and cannot simultaneously communicate with all robots in the global range, indicating that the system is distributed[3]. It is important to note that in swarm robotics, *local* is relative, depending on the size of the robot population, the system, and the goal. The motivation for suggesting local communication is the need for autonomous robotic systems to be more robust.

The robotic system I built adheres to the above characteristics and constraints of swarm robotics. The advantageous properties that these principles are theorized to induce are discussed next.

Reason for using swarm robotics

The abstract principles or model of Swarm robotics can be seen in several animal populations. If a single creature cannot execute a job on its own, but can partner with other organisms and perceive environmental information, it is able to overcome complicated obstacles and complete challenging goals[7]. Similarly, in the robot system, a single robot may not be able to perform a complex mission; nevertheless, via communication and collaboration with neighboring robots in the group, the robot system may exhibit a range of behaviors and acquire distinct features. Swarm robotics pursues a way to design for this multi-robot collaboration and collective behaviours without a central organising authority.

As Wolframe said[8],*"The complexity is generated by the cooperative effect of many simple identical components."* It is theorised that using multiple basic individuals provides a set of unique

characteristics to Swarm Robotics that allow for their flexible use in a range of circumstances. These are explore further below:

- Simplicity: Due to Swarm Robotics' capacity to reconfigure cheaper and simpler robot units dependent on the task at hand, teams of basic robots may do complicated tasks more effectively than single multi-functional robots. Due to the simplicity of the robot's individual components, programming mistakes are infrequent, and individual performance can be effectively debugged and assessed[9].

- Scalability: As the base units in a swarm robot system all have the same function, play the same role, and are able to determine their own behaviors based on information obtained from local communication, this suggests that the system can use its self-organizing to function properly even if the size of the robot population is increased. So, depending on the difficulty of the task and the scenario, the robotic system can be scaled up or down, keeping the robot density constant[10].

- Robustness: Due to the high level of redundancy within the swarm robot system, even if a single robot fails, the whole robot system will not be paralyzed, since the other robots have the same responsibility and functionality[11]. Because the system does not reliant on a central authority to arrange each robot, the failure of a single robot will not result in the loss of essential functions.

- Flexibility: The distributed, self-organizing, and decentralized characteristics of swarm robots help them to work flexibly in different operating environments and locations: individual robots in a swarm distribute tasks dynamically for themselves through information obtained by local communication. Since the robots do not need to communicate on a global level but rather acquire information through local sensing and communication, they can work in environments that are not equipped with high-tech infrastructure.

- Parallelism: Since robots are autonomous and can decide their next actions by themselves, and swarm robotics is redundant, the system can achieve the effect of multi-task parallelism provided by the aggregation of a large number of coordinating robots.

- Economical: The simplicity of swarm robots makes them far less expensive to construct, develop, and maintain than individual robots with complex functionality, even if swarm robots should be manufactured in huge quantities. Individuals in a swarm could be produced on a vast scale, but robots with complicated tasks need more sophisticated processing[12]. This also makes it easier to repair broken swarm robots and less expensive to replace them directly when they are seriously damaged.

In summary, swarm robots are intended to allow basic robots to interact with each other and hence expand the system's potential by increasing robot density and generating complex behaviors from simple agents, as opposed to explicitly coordinating (through programming or otherwise) the hardware capabilities of individual robots. In comparison to conventional robotic systems, swarm robots is proposed to provide superior robustness, flexibility and scalability, autonomy, simplicity, and cost-effectiveness.

The benefits of swarm robots are especially obvious in situations when the environment is harsh, and likely to harm the robot, since the cost of repairing the robot is affordable, or swarm robots are low-cost enough to simply replace. Furthermore, adding or subtracting robots from the system should not impact the collective performance. In addition, the theory of self-organization[13] suggests that ambiguity in the behavior of individual agents may sometimes considerably enhance the system's capacity to experiment with potential solutions (for example, behavioural deviation in path following allows ants to optimise for shorter paths to food sources).

This study is intended to contribute to the field of swarm robotics, so the study employs short range communication and explores what utility can be extracted from relatively simple robots. For instance, if this were a multi-robot system, a centralised control strategy could be employed. After discussing these advantages of swarm systems, I will next illustrate the utility of Swarm Robotics by discussing its applications.

Applications of Swarm robotics

To develop a system without centralized control, local sensing and communication will be essential for task completion. In order to differentiate between local and global communication, I will now define the various types of communication and perception.

The three types of local communication are *interaction via environment*, *interaction via sensing* and *interaction via communications*[14]. Among them, interaction via Sensing, and interaction via communications are the communication modes possible as a direct channel between robots, whilst the third is an indirect method of communication which must take place by robots manipulating their task environment:

- Interaction via Environment: Robots often interact with their surroundings by altering the environment and leaving clues, which other robots could use to determine the state they should enter and their next action to take. This technique is often used in systems using finite state machines, which is a kind of implicit communication, and is also known as *stigmergy* or *stigmergic communication*[15].

- Interaction via Sensing: Interaction via sensing is the interaction that happens when robots perceive each other but do not communicate openly and directly. Such communication enables robots have the ability to differentiate between robots and other objects. This may include, for instance, perceiving the action (i.e. movement) of another robot as opposed to getting an explicit message of actions from the robot.

- Interaction via Communications: Such communication allows specific and accurate messages to be exchanged between robots, while the recipient of the information can be either specified or unspecified, which has contributed to the growth of communication network architecture and protocols.

Within direct communication, there are three ways based on different communication ranges[16]:

- Private communication: Communication only occurs among selected agents.

- Local communication: Communication only occurs among neighbors within the surroundings, either through interaction via environment, sensing other robots, or communication technologies within a relatively short range. In this category, communication is normally inclusive to neighbours and does not discriminate (e.g., is not private).

- Global communication: All agents in the system can communicate with one another. But this strategy is not considered for usage in the Swarm Robotics system, since it would eliminate

several of the system's advantageous qualities, such as its flexibility and economy. This is because it would become possible for collective behaviours to either be designed or to emerge with a reliance on information from the whole system - and this would cause a potential point of failure for a swarm system that is expected to adapt to robot failures or significant changes to the task environment.

Another categorization of communication is based on the message's substance. Balch and Tucker examined the effects of no communication, state communication, and goal communication[17]. Their study focuses on the foraging challenge, where robots must find items and retrieve them. The conclusion is that in the foraging challenge goal communication (announcing the position and direction of the target object) had superior performance compared to state communication. In my project, I will only utilise no communication and state communication because of the limitation of communication board and evaluate both methods by comparison, then judge the necessity of the goal communication.

Nathan and John's[18] socially-inspired communication approach allows the robot to determine whether to receive a message depending on the message's relevancy. The relevancy of the message is depending on the moment of its transmission. The earlier a message is sent, the less relevant it becomes (or in other words, a message can become old). This strategy guarantees the immediacy of the communication, but imposes stringent technological constraints on the physical robot, since all robots must be able to synchronize the time. In my project, I will make the robots update their messages for sending every 100 milliseconds, to avoid outdated message being received by other robots.

Hoff introduced two ant-like algorithms for swarm robots to coordinate their behaviour during foraging that rely on basic local communication which need the use of stationary robots as beacons to mark out the route from the food source to the nest[19]. This method of using beacons is a technological alternative to the way-finding of pheromone trails used by biological ants. In this project, this algorithm is not considered because there are many paths to push the box to the target location, box locations will change quickly, and the method is not technologically feasible with the available resources. As such, the box-pushing task can also be taken as a selection of task that is suitable for the available technological resources. The box-pushing task provides similar challenges of locating and manipulating objects, but without a significant challenge of localisation

and navigation. The design of the box-pushing task for this study is discussed further in section 2.1.

In Sina's paper[20], the phases of cooperative robot transport are: decision phase, recruitment phase, organisation phase and transport phase. This study adopts similar states in the finite state machine implemented: an exploration state, a judgement state, a recruitment state, and a push-box state - further information can be found in section 2.2.1. Sina's results showed that the task of cooperative transport of objects can be achieved by swarm robots without communication and perfect sensors, and is well scalable. The author mentioned that adding explicit local communication might further improve performance. This project therefore helps to advance knowledge in the research field by including explicit communication. This is achieved through the use of an IR-based communication board, more details can be found in section 2.3.7.

This article by Ahmad et al.[21] introduced a new concept: intention recognition, which differs from plan recognition in that it can even identify actions that the observed agent has not yet performed, i.e. plan recognition only focuses on the present, whereas intention recognition can identify the present and predict the future. Intention recognition has also shown great benefits in simulated box-pushing experiments, but its application to a physical robot may require the robot to have vision capabilities, which the robot used in this project does not currently have. However, I use a similar implicit communication scheme to facilitate the box-pushing task and compare this to explicit communication. Taking inspiration from Ahmad et al, this work investigates a type of object recognition where robots in the system can use the information it obtained by this functionality to determine its next action. More information is available in section 2.3.6.

Li, Shuai, et al. [22]in their study concluded that global communication is superior than local communication while looking for light sources. This project will remain based on local communication to adhere to the swarm principles. However, a key discovery of Li, Shuai, et al. was that explicit communication leads to improved performance. As such, this project will further knowledge by evaluating both explicit and implicit communication schemes for impact in the proposed box-pushing task. I hypothesise that having too many robots in a small spatial region may be harmful to the effectiveness of box pushing (i.e. robots will cause obstructions and barriers to one another. Therefore, I hypothesise that local communication may be employed to better coordinate the robots to mitigate these inefficiencies. How and whether this hypothesis is true is the scope of

the experimental investigation of this study, detailed further in chapter 5.

Challenges in Swarm robotics

Even though Swarm Robotics has the good characteristics mentioned in section 1.1.1, it may still be harmed by malfunctioning robots. For instance, a defective signal receiver or sensor may lead the robot to acquire incorrect information, while a broken signal transmitter could cause the robot's neighbors to receive incorrect signals. Even though the communication is local, if the issue is not identified in a timely manner, the whole system may be impacted constantly and the mission may fail[10]. Along these lines, this study will report findings on the general performance of the system across multiple communication strategies, with an aim to reveal both strengths and weaknesses.

Due to the uniformity of Swarm Robotics, it is not very difficult to program individual robots in Swarm, and all robots share the same code. The challenge is in ensuring that the robot interacts with a local neighborhood or environment that prompts the appropriate group behavior. Group behavior is the superposition of several individual robot actions, which is characterized by complexity and unpredictability. On the one hand, we may anticipate that Swarm will do well with the simple agents' interactions, but on the other hand, it is difficult to predict the result until we conduct the experiment.

Even though the Swarm Robotics system can operate normally with less manual intervention than traditional technology due to its robustness, scalability, and flexibility, as stated previously, it is still necessary to maintain the system if there are faulty individuals in the system or if the communication is unreliable due to destroyed hardware, leading to incorrect information transmission. Understanding how to determine whether to do maintenance and how to do maintenance on swarm robot systems requires more investigation.

Cooperative robots should be capable of communicating with one another in order to obtain information outside their own perspective. When a group of robots is looking for a target, for instance, the robot that locates the objective will broadcast its position with the other robots. Therefore, reliable and robust communication between robots is essential and will aid in the development of coordination algorithms. In this project, the communication efficiency will also be evaluated.

Regarding to the challenges mentioned above, this project will explore the issues outlined while developing robotic systems and their communication strategies, and the project's result will com-

prise an awareness of the challenges and the solutions discovered.

1.1.2 Examples of Box-Pushing

The proposed robotic system by Gerkey[23] employs a heterogeneous model: there are roles of pusher and observer, executing the tasks of pushing the box and observing the target position, respectively, and communicating through WIFI. Considering that it may not be possible to deploy WIFI in harsh working environments, I will use local infrared communication applied to each robot in my project so that the robotic system can communicate without the environment providing additional conditions, allowing it to be adapted to a variety of scenarios. Furthermore, I create an implementation that is more aligned to the characteristics and design constraints of swarm robotics, and look to contribute a study of box-pushing into the field of swarm robotics research.

Kube and Bonabeau [13] have constructed a collaborative box-pushing mechanism based on the cooperative transport of ant colonies. The robots in this system communicates indirectly, also known as implicit communication, i.e. it relies on cues in the environment or perceptions of other robots as a basis for making judgements about its next move. Experiments have demonstrated that it is feasible to build cooperative robot behaviour using implicit communication alone. In this study, I will likewise implement implicit communication in my project and assess its performance. Furthermore, in this study, I will advance knowledge by including explicit communication schemes. The study by Kube and Bonabeau also examines the size of the object being pushed; the larger the box the more touchable areas, the more robots can be engaged in the task of pushing the box, and the more successful the task will be. For my project, I will design the box to be a size that at least two robots can push simultaneously, in order to better represent whether cooperative behaviour is generated. In contrast, a box small enough to be pushed by one robot does not require cooperation.

Ghosh's research[24] used an improved multi-objective particle swarm optimisation algorithm to solve the problem of multiple robots cooperating to push a box, with implicit communication used by the system. It also uses a local optimum approach to determine magnitude of the force applied by two robots, with the force applied being related to the current box position. Within my study, there is a significant technical challenge: a relatively simple robot must be able to discern the difference between pushing another robot or pushing a box. My project investigates the use of feedback from PID controllers used to create consistent wheel speeds for the robot. The feedback

from these PID controllers are used to infer the forces applied by the robot engaged in pushing. Further information can be found in section 2.3.6.

1.2 Summary & Contributions

The above literature review has covered ideas and concepts to enable cooperation amongst robots in swarm robotics. Concerning aspects of communication, explicit and implicit communication have been advocated in particular. Since implicit communication does not need much technology and could be rather effective, more and more study is being conducted on how to transmit messages by sensing and altering the environment to leave cues, which is a concept known as 'stigmergy'[15].

As seen by several studies on implicit communication, researchers have hypothesized that explicit communication might improve system performance, and some have even proposed that the greater the number of messages exchanged, the better the system performs. This encourages me to believe that local explicit communication should have certain benefits. To investigate this point, I will compare robotic systems using distinct communication techniques.

This project will also explore the challenges and constraints of existing swarm robotics based on the relatively incapable robots utilized and attempt to address them, such as identifying the type of object the robot is pushing, in an effort to develop technologies that will improve swarm robotics.

The two main academic contributions of my work are:

1. The IR-based communication board is used to investigate the impact of local explicit communication in a cooperative box-pushing task and to compare the benefits and drawbacks of explicit communication with implicit communication. Therefore contributes to the advancement of knowledge in swarms by including explicit communication.

2. In my project, I will employ very basic robots to overcome significant technological obstacle: robots without visual capabilities and explicit communication skills will be able to distinguish between a box and a robot solely on PID feedback. This provides a substantial contribution to robotic systems comprised of very fundamental robots.

1.2.1 Research Questions

This project aims to construct a robot system that can efficiently fulfill the job of box pushing utilizing a less expensive robot. Cooperation requires effective communication. I will employ local communication, which can be subdivided into implicit communication and explicit communication, to achieve efficiency and high adaptation to varied contexts. The two broard research questions I propose are:

1. to investigate are how much local communication, and what types of local communication, can improve the collaboration.

2. to investigate which elements of a robotic system have a significant influence on local communication and it's application.

In the publications about swarm robotics collaboration, some researchers did not adhere to the Swarm Robotics principles, while others compared just one communication mechanism to the baseline. My study subject will keep a strict adherence to the aforementioned principles in section 1.1.1 in order to assess the benefits and drawbacks of different methods of communication.

1.2.2 Aims & Objectives

This research will examine the effects of local communication when robots collaborate to accomplish tasks. The selected task for the project is the push box task, a classic kind of object transportation found frequently in everyday life. Object transportation is also known as the 'piano-mover problem'[25], in which it is essential to not only move the object to the target location, but also expect the robots to cooperate well, maintain close contact with the target object, and attempt to save task time by transporting the object in a linear motion. To make robots cooperate, the communication among the robots is indispensable.

As for the principle of swarm robots mentioned in the previous section, I will adhere to the use of local communication rather than global communication without destroying the flexibility, scalability and adaptability brought by this communication to the robot system. As Mohamed et al.[26] found, the range of communication is negatively related to group fitness. Therefore, I am primarily concerned with which local communication technique may reduce costs while maximizing benefits

and efficiency, and what factors can increase the percentage of task completion success. The aims and objectives are listed below.

Aims

For this project, my main direction and purpose is:

- First, I want to know to what degree local communication might boost the cooperation impact of swarm robot collaboration and which communication strategy offers the most benefits.
- Second, I want to determine which specific communication aspects have the major influence on communication and if portraying clearer information is more advantageous for robot collaboration.

While investigating the topics that interest me, there are challenges in Swarm Robotics that necessitate being resolved. Solving these difficulties also enhances the functions required by this system, which enables me to get a deeper understanding of what I want to explore. The challenges are:

- Is the complexity and additional cost of introducing a communication mechanism to a robotic system worthwhile? This question is central to the whole research, and only if communication is valuable will the investigation of other concerns be meaningful. It may be hard to conduct a quantitative study of this problem in its entirety, but the trend of the gathered data may provide some insight into this issue.
- What is the most effective communication method? Balch stated in his research[17] that goal communication is superior than state communication in the foraging task. I want to determine if goal communication is required or whether merely state-based communication could be effective enough. This paves the way for enhanced explicit communication.
- How can a robot determine if the thing it is pushing is its target object? In order to complete the objective, the robot must push the box, but it is quite possible that it will encounter another robot instead. Therefore, the robot should demonstrate good judgment to avoid spending its limited time on worthless actions. My early guess of the idea solving this challenge is to

utilize PID feedback to detect the type of object since the box could be gently pushed away but robots equipped with PID will engage in combat which will be indicated in PID feedback.

- How do the robots coordinate their movements in order to facilitate efficient box pushing? By simply choosing a constant forward direction when pushing the box, the position of the robot's force points and the interference of other robots when pushing the box can easily cause the box to deviate from the trajectory the robot wants to take, resulting in the robot becoming separated from the box and the robot having to explore the arena again to find the box. It is therefore essential to develop an algorithm that gives the robot better control over the box.

- How to solve the problem that other robots getting in the way? When a robot is pushing the box, other robots blocking the way will not only slow down the process of pushing the box, but may also cause the box to be out of the control of this robot, forcing the robot to search the box again. So it is necessary to reduce the occurrence of this situation.

Objectives

The objectives of this project is listed below:

- Program individual robots to push boxes. Try to get the robot to push the box out in a straight trajectory. This is the most basic behavior that the robot needs to be able to achieve in the whole project.

- Design the shape and size of the box, set up the experimental arena, and determine the size and smoothness of the experiment arena's surface. These values can be tested by robots with the capacity to push boxes.

- Implement an algorithm to utilise feedback from the PID speed controllers, used to maintain constant wheel speeds, to allow the robot to discriminate between pushing a box or pushing another robot.

- Program the robot to send and receive messages. Investigate and design communication strategies between robots.

- A set of studies, including a baseline experiment, an experiment on implicit communication, and an experiment on explicit communication using various communication styles, should be developed. It includes predicting robot behaviors worth recording, collecting experimental data, analyzing experimental results, etc.

1.2.3 Hypotheses

Following the aims and objectives, my first hypothesis is that:

- It is possible for a robot to discriminate between pushing a box and pushing another robot, by using the feedback from PID speed controllers. This functionality is essential for the task of cooperative box-pushing, where a robots should not spend time pushing each other. This hypothesis operates on the theory that a box can be designed to be lighter, and therefore easier to push than another robot. Furthermore, for a given desired forward speed, a robot will be able to recognise when the feedback has reached higher thresholds, and that these feedback levels will be different for unobstructed travel, box pushing, and pushing a robot. This hypothesis is investigated and documented in section 2.3 (implementation), as it is not the primary investigation of this project.

- To realize that the robot can control the forward movement of the box, the robot should first alter its own forward direction and forward velocity in order to exert force on the box. thus hypothesize that a robot equipped with bump sensors, a PID speed controller, and a PID heading controller may attain this outcome. The PID speed controller can maintain the forward velocity of the robot. The robot processes the readings from the bump sensors into the required turning angles before using the PID Heading Controller to seamlessly regulate its steering. This hypothesis is investigated and documented in section 2.3

The primary experimental investigation of this project has the following hypotheses:

1. My initial hypothesis on the aforementioned research questions is that local communication can really help swarm robots cooperate with each other and thus improve task performance, and that explicit communication is likely to be more effective because the messages conveyed by explicit communication are more unambiguous. While robots with implicit communication cannot get to know the state of other robots and then join others' box-pushing behaviour.

2. Regarding the factor that is most important to the task's success rate, I hypothesise it may be the number of times that robots successfully collaborate, since this should considerably increase the speed of pushing the box and the straightness of the trajectory when pushing the box. As can be seen in figure 1.1, when there is one single robot pushing the box, the shape of the trajectory may be a circle, but when two robots collaborate it is more possible that the box can be pushed straight forward thus reduce the length of the path(aim of the task is to push the box out of black line).

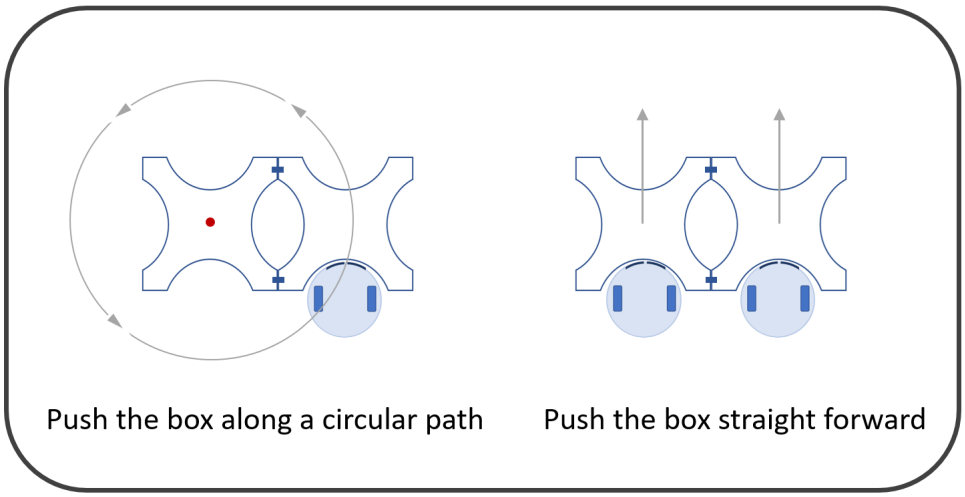


Figure 1.1: The trajectory of the box

3. I hypothesise that explicit communication may facilitate the formation of collaborative behaviors in a timely manner, and therefore be more efficiently overall. For example, an explicit communication can travel distance between robots nearly instantaneously, whilst implicit communication requires robots to first encounter one another physically.

2 Research Methodology

This chapter describes the framework for conducting the experiment investigation. The experiment can be summarised as investigating whether local communication strategies can improve the performance of swarm robots pushing a box out of the enclosed arena. This task is elaborated here section 2.1. In order to compare the impact of different communication strategies, I take the non-communication mode as the baseline and set up three different communication strategies to evaluate by comparison. These three strategies are implicit communication, explicit communication: recruitment and explicit communication: halt. These communication strategies are detailed in section 2.2.1. After discussing the implementation of the system, this chapter details the design of the experiment methodology through section 2.2.1, including specific parameters and the procedure for the part of the experiment that required human participation.

2.1 The Box-Pushing Task

Box-pushing task is, multiple robots in an arena and try to push the box out of the black line(seen as figure 2.1). Once the box is pushed out, this will be considered as a success. In evaluating the effectiveness of each method of communication, the number of completed tasks will serve as the primary indicator. The construction of the experimental scenario and the box design for this task will be introduced in detail in section2.3.8.

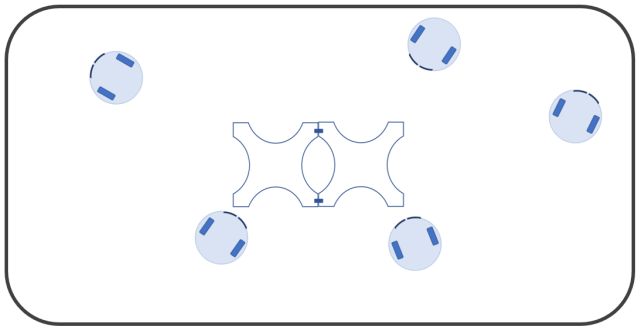


Figure 2.1: Scenario of task

2.2 System Overview

In order for a robot to perform the task of pushing a box, the robot's behaviour needs to consist of a number of specific actions, such as searching for the box, pushing the box, sending and receiving messages, waiting for help, etc which will make up the complete robotic box-pushing system. This section will describe the box pushing system under different communication strategies, clearly illustrating the relationship between the various actions and decisions.

2.2.1 System with different communication strategies

No communication system

When there is no communication throughout the robots' behavior, it is essential to note that neither explicit nor implicit communication will occur. In this system, the robots will just do Brownian motion[27] which constitutes a type of randomised traversal of the task environment. The robots will not know when to push the box (based on cues in the environment), and they will not utilize the communication board to communicate with other robots. So they can't sense the position of the box and deliberately push it, the box is pushed only because the robot needs to move forward. Therefore, the baseline can be considered to produce a level of performance without any intention or task-solution programmed into the robots at all.

When the robots do the Brownian motion[27], the robot randomly change its heading direction every 800ms. When the robot is in Brownian motion, the angular variation in the forward direction (radian system) is normally distributed with a mean of 0 and a variance of 0.5. To control the robot to the required angle, I will use the kinematics function to estimate the angle that the robot has already turned to find the angle that still needs to be turned and use PID to control the speed of the two wheels to converge the directional error to 0. A detailed description of this function will be provided in section 2.3.5 and 2.3.6 when describing Kinematics and PID.

Implicit communication system

Implicit communication involves interaction via environment and sensing. In this experiment, the system is shown in the figure 2.2. If the robot collides with a box while performing Brownian

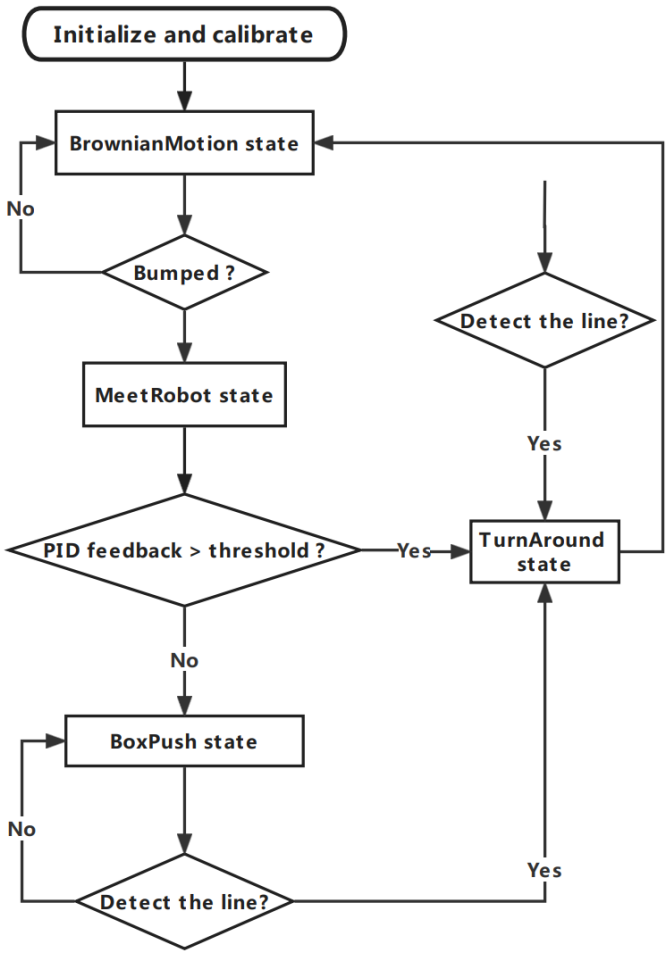


Figure 2.2: Implicit communication system

motion[27], it goes into the state of pushing the box in which the readings of left and right bump sensors manage the speed of the wheel rotation, allowing it to push the box along a straight line. The robot would turn around when it detects a black line. The implementation of box pushing in a straight line and *TurningAround state* will be described in detail in section 2.3 when introducing bump sensor, line sensor and PID.

Under this system, the robots explore the environment and interact with the environment only. Therefore, a robot can change the environment through it's own interaction with the environment, and another robot can perceive the condition of the environment as a cue for behaviour. So other robots choose their next action by the cues in the environment, which is a kind of implicit communication.

Explicit communication system: recruitment

Using explicit communication, this kind of engagement enables the robot to communicate transparently and convey more specific data. The flow of this system is shown in the figure 2.3.

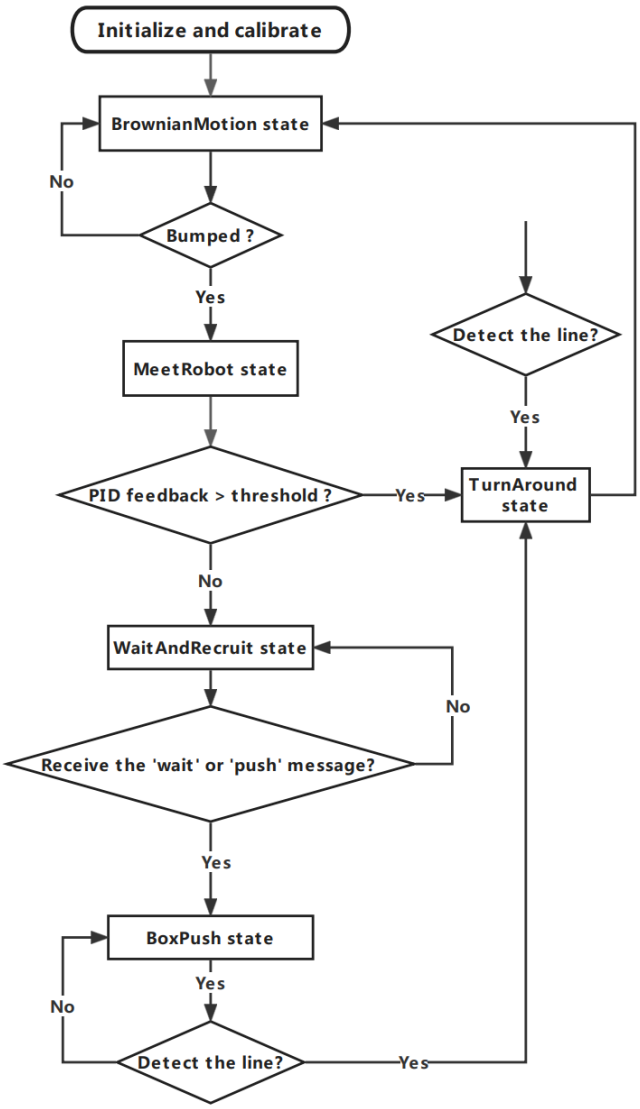


Figure 2.3: Explicit communication system

In this system, a robot always sends out a message containing its current state every 100 milliseconds and receives a message from outside every 150 milliseconds using communication board. If a robot realise that it is bumped, it will use the PID feedback to determine the type of the bumped object - this is detailed further in section 2.3.6.

If the feedback is smaller than the threshold, then it means the robot encountered a box and

will go into the *waitAndRecruit* state. When a robot is in the *waitAndRecruit* state and receives a message that another robot is also in this state or in the *BoxPushing state*, then the robot will also go into the *BoxPush state* to start pushing the box with its companion. During the whole loop, whenever the robot detects a black line it will turn around.

Explicit communication system: Halt

This type of communication is similar to the process described above for basic explicit communication. The only difference is that when one robot start to push the box, it will send message to tell the robots which are not in *waitAndRecruit state* or *BoxPushing state* to stop at the same place for 2 seconds. If a stationary robot continues to receive messages from a robot that is pushing the box, it will remain stationary. In that way, I expect the robots which are around the ones doing box pushing task will not block their way to interrupt their pushing behavior.

2.2.2 Communication Strategies

1. **no communication (NC):** No communication means neither communication using communication boards to transmit messages to other robots, nor perception of the environment using sensors to obtain cues in the environment. The only sensor that robots use is line sensor which prevents the robot from going out of the experimental arena.

2. **implicit communication (IC):** Unlike expicit communication, implicit communication refers to the transmission of messages to other robots via the environment. This communication strategy is implemented through the bump sensor in this experiment. When the readings of robot's bump sensors feed into PID and returns a feedback indicating that it has collided with an object, it enters a state in which it identifies the type of object it has collided with, and this is one of the most important functions of implicit communication.

   If it determines that it has collided with a box, the robot gets the readings from the left and right bump sensors and try to minimize the difference between the two readings with PID function by adjusting the the robot's heading direction to make the surface of the box apply the same force to the left and right skirts of the robot's bumper. This process is shown in the figure 2.7. This function is also an important part of implicit communication.

3. **explicit communication, recruitment (EC-R):** Using explicit communication, robots need to utilise I2C extension boards which provide local infra-red (IR) communication to send and receive messages.

   Every 100 milliseconds, a robot transmits a message with its current state. When a robot in the *waitAndRecruit state* gets a message indicating that another robot is in the same state or the *BoxPushing state*, it will switch to the *BoxPushing state* as well. Since infrared message is unable to penetrate the barrier, I chose a box that is slightly taller than the robot so that if the robots are not on the same side of the box, they cannot receive each other's messages, which prevents them from pushing in the opposite sides of the box each other to some extent.

4. **explicit communication, halting (EC-H):** This kind of communication is based on EC-R. When a robot enters the *BoxPushing state*, it broadcasts this state to its neighbors, and the other robots which are not in *BoxPushing state* or *WaitAndRecruit State* halt moving for two seconds when they get this message(if they continue to receive the message they will stay in place). After finishing the *Stop state*, the robot returns to the *Brownian motion state*. The pseudocode for this section is shown in algorithm 1 below:

Algorithm 1 explicit communication, halting (EC-H)

```
value_received = receiveMessage()
if value_received == STATE_BoxPush then
  if (state == STATE_BoxPush)||(state == STATE_WaitAndRecruit) then
    state = STATE_BoxPush
  else
    state = STATE_Stop
  end if
else if value_received == STATE_WaitAndRecruit then
  if state == STATE_WaitAndRecruit then
    state = STATE_BoxPush
  end if
end if
```

2.3 Implementation

In order to explore the influence of local communication on robot cooperation, I will use 3Pi robot (as shown in the figure 2.4, available from [28]), a relatively simple and incompetent robot, as the basic unit of the swarm robot system. The objective of the 3Pi robots is to push a single box to the outside of the area bounded by the black line. This requires the robots to have certain internal and external functionalities to help them complete the complicated tasks, such as distinguishing pushing a box from pushing another robot, pushing the box, local implicit and explicit communication.



Figure 2.4: Pololu 3Pi+ mobile robots

Figure 2.5 demonstrates the components that enable the robot to have these functions, which are described in detail below.

An overview of these functionalities is provided:

2.3.1 Software Architecture

During the box-pushing task, the robot will cycle through many phases of operation, such as: traveling around in search of boxes, pushing the box, waiting for help, sending and receiving messages, working with peers, etc. Thus, when the complexity of the operations to be performed grows, it becomes necessary to build a finite state machine to simplifies the code and expedites the development process[29]. The specific workflow of each communication methods is demonstrated in section 2.2.

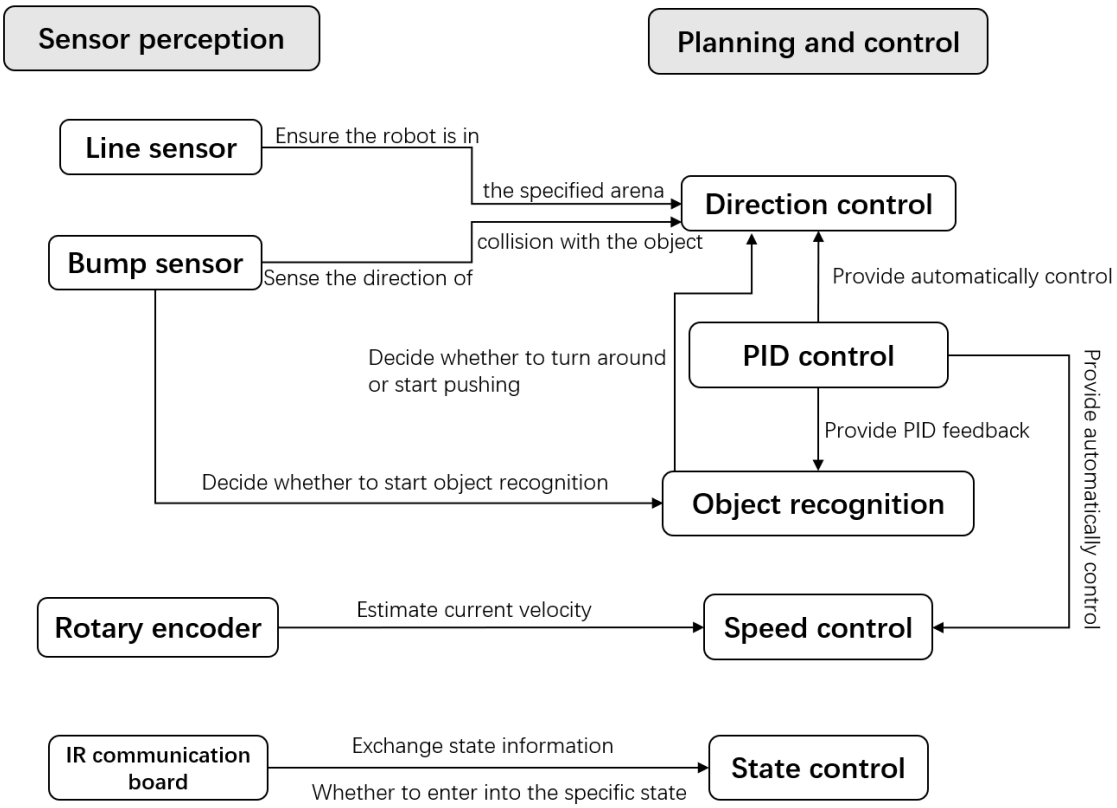


Figure 2.5: System components

2.3.2 Line sensor

The line sensors are positioned beneath the robot and they are used to allow a robot to detect the edge of the arena. The five line sensors are located underneath the front of the robot(shown in figure 2.6), they are all reflectivity sensors and they only provide a simple reflectivity reading to determine how light or dark the surface is underneath the robot[30]. The higher the value returned by the sensor, the darker the surface. In this project, when the line sensor detects a black line (i.e. the edge of the area), the robot will turn to get back inside the area and thus continue searching for the box.

I set the turning_around movement to make the robot move backwards three centimetres to control the speed of the motor and then rotates 180° so that the initial direction before entering the next state would be towards the inside of the arena. This action is also performed when the robot recognizes that it has collided with another robot, and the action of stepping back leaves room for

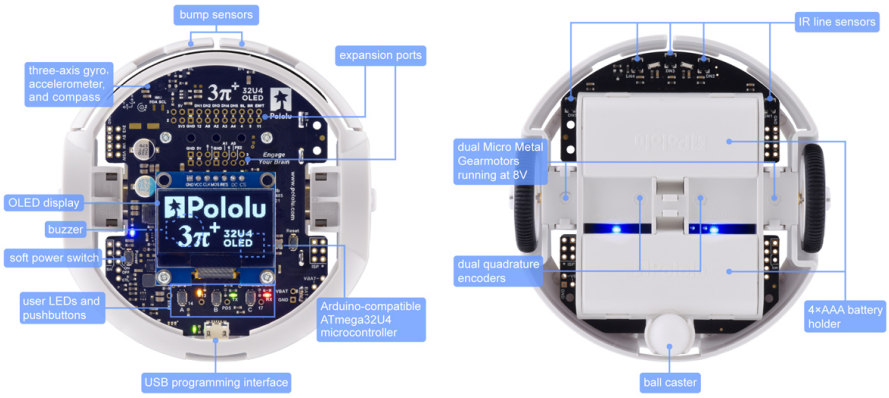


Figure 2.6: 3pi robot features(available from [28])

the robot to turn around smoothly.

2.3.3 Bump sensor

The Bump Sensor is comprised of two reflectance sensors and a bumper skirt on the front of the robot(shown in figure 2.6), which determines if the robot's front has bumped with other objects.

Unlike line sensors, the reflectance sensor measures changes in reflected light, so when the flap in front of the 3Pi's bump sensor is pressed down, the 3Pi detects whether the contact is on the left or right side. Since the robot is able to determine the position of the object it is colliding with, it can modify its direction to push the object in a straight line (for example, if the robot senses that the object is on the left, it will speed up the right wheel, seen as figure 2.7).

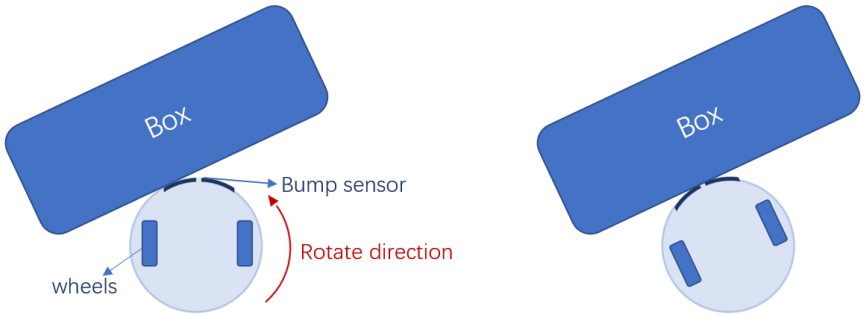


Figure 2.7: Direction ajust

Because bump sensors could detect changes in reflected infrared light, they are very susceptible to other sources of infrared light in the environment[30]. Therefore, I put in a calibration procedure

to eliminate that effect. After the robot is turned on, I press its bumper skirt twice, and at the same time the robot collects the sensor readings during calibration to obtain the maximum and minimum readings, based on which the subsequent readings will be processed as following equation:

$$reading = \frac{reading_orignal - reading_min}{reading_max - reading_min} \tag{2.1}$$

Regarding equation 2.1 above, *reading_original* refers to the latest bump sensor acquired by the robot, *reading_min* and *reading_max* is the minimum and maximum value of the readings from calibration.

2.3.4 Rotary encoder

To keep track of how many times the wheels have spun, every wheel is fitted with a rotary encoder, which consists of two hall sensors arranged at right angles to one another. When the wheel is in motion, the magnet follows suit, spinning to create a magnetic field around the sensor circuit and triggering a number of different signal outputs. The interrupt service routine(ISR) is activated whenever there is a change in the signal outputs[29].

With the rotary encoder, the current speed of the robot can be easily obtained. For each loop of the system, the cumulative number of wheel revolutions enc_count is counted, and the average speed over this time is obtained by the number of wheel revolutions dividing the time difference between the two loops which is shown in the equation 2.2 below. Since each loop is separated by a very short time, this average speed can be approximated as the current speed.

$$speed = \frac{enc_count - enc_count_previous}{millis() - spd_update_ts} \tag{2.2}$$

Regarding the equation 2.2 above, *enc_count_previous* is the number of wheel revolutions obtained in the last loop, $millis()$ gets the current time spot and spd_update_ts refers to the time spot when the speed was last calculated.

2.3.5 Kinematics

Using kinematics, the pose of the robot relative to its initial starting point can be estimated[29]. To achieve this, every short time Δt the robot calculates the angle θ and the distance traveled toward

the x and y axes (X_R and Y_R, respectively), summing these small angles and distances to get the cumulative distance traveled and the angle of rotation. The equations to calculate $\dot{X}_R$ and $\dot{Y}_R$ are shown in equation 2.3

$$\begin{aligned} \dot{X}_R &= \frac{r\phi_l}{2} + \frac{r\phi_r}{2} \\ \dot{Y}_R &= 0 \end{aligned} \tag{2.3}$$

The parameters in the equation are shown in the figure 2.8. X_I and Y_I are the axes of global

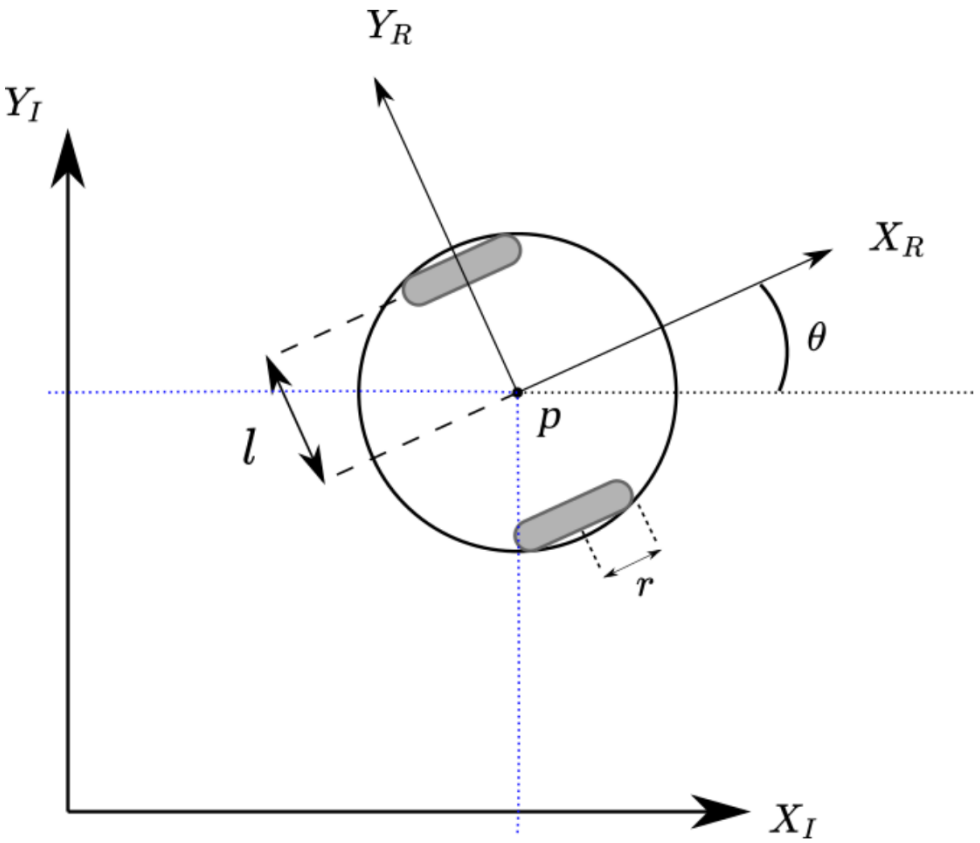


Figure 2.8: Robot kinematics[29]

co-ordinate frame, the origin of this frame is the position where the robot is powered on. X_R and Y_R are local co-ordinate frame, so the position of the wheel with respect to the center of the robot can be represented. r refers to the radius of the wheel and l is the radius of the robot. ϕ_l and ϕ_r is the Angle at which the left and right wheels of the robot rotate respectively, these could be obtained from rotary encoders.

Since the robot is always going in the direction of X_R, $\dot{Y}_R$ is always equal to 0. And $\dot{X}_R$(the distance traveled by the robot) is calculated as the average of the distance traveled by its two wheels

as equation 2.3.

To understand the robot's current position, the most important thing is to calculate the $\dot{\theta}$, Angle of each rotation of the robot. This angle can be obtained as the difference in the angle at which the contribution made by each of the left and right wheels causes the robot to turn. The equation to calculate it is shown in 2.4.

$$\dot{\theta}_R = \frac{r\phi_l}{2l} - \frac{r\phi_r}{2l} \tag{2.4}$$

So the coordinates and posture of the robot under the global co-ordinate frame can be obtained by equation 2.5. t is the number of iterations.

$$\begin{aligned} X_I^{t+1} &= X_I^t + \left(\dot{X}_R \cos\theta_I^t\right) \\ Y_I^{t+1} &= Y_I^t + \left(\dot{X}_R \sin\theta_I^t\right) \\ \theta_I^{t+1} &= \theta_I^t + \dot{\theta}_R \end{aligned} \tag{2.5}$$

2.3.6 PID controller

The PID controller can adjust the input value of the system by converting the error signal into feedback signal to achieve more accurate and stable control of the system[31]. The PID controller works as shown in figure 2.9.

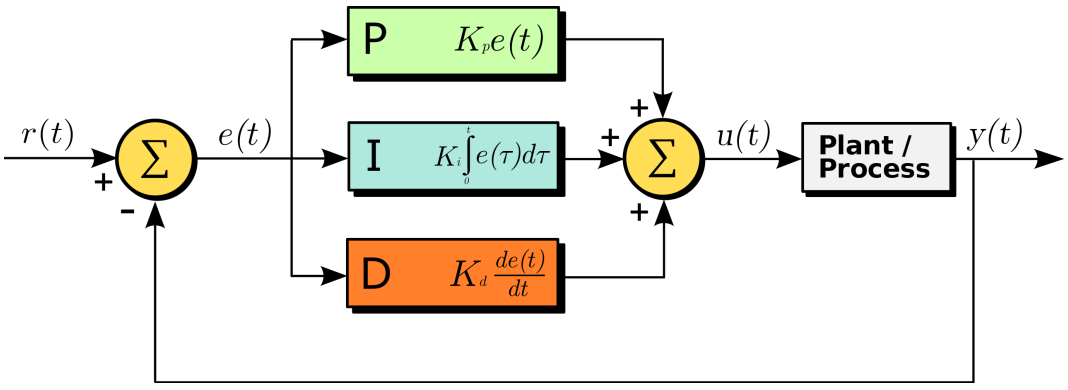


Figure 2.9: Simple illustration of a PID controller, image from Wikipedia [31]

The input is $r(t)$ and the output is $y(t)$, $e(t)$ is error which is equal to $r(t) - y(t)$. $u(t)$ is the feedback signal which is used to adjust output to the demand value(input). The gain of P, I, D which is K_p, K_i and K_d are initialised as specific value based on scenario requirement. The calculations of

P, I and D are shown in equation 2.6.

$$\begin{aligned} P &= K_p * error \\ I_{sum} &= I_{sum} + error * \Delta t \\ I &= K_i * I_{sum} \\ D &= \frac{K_d * (error - last_error)}{\Delta t} \end{aligned} \tag{2.6}$$

And the PID feedback is the sum of P, I and D shown in equation 2.7.

$$feedback = P + I + D \tag{2.7}$$

In this project, the PID controller primarily assists in the performance of three functions: direction control, speed control and object recognition.

Direction control and Speed control

When pushing the box, the different pressure applied by the box to the left and right bumper skirts of the robot causes the bump sensors to read differently. In order to keep the robot applying a force perpendicular to the surface of the box, it needs to keep the bump sensors readings the same. So for the direction control feedback, the input is 0 (the demand difference between the bump sensors' readings is 0) and the measurement is the difference between the two bump sensors' readings. After getting the direction control feedback, it will be used as a factor to adjust the demand velocity of two wheels for speed control.

Object recognition

The PID feedback enables the robot to identify the object it bumped. By comparing the PID response of a robot traveling on a flat surface with no obstacles, the PID feedback allows the robot to differentiate between pushing against another robot and pushing a box. When the robots encounter another robot, they both adjust their speed to try to push each other and the result is that they are stuck in place. two robots both pushing each other with contact on their fronts is basically equivalent to a robot pushing a stationary object, and the PID feedback will increase rapidly. The PID feedback when two robots fight each other is shown in the graph 2.10. From the graph 2.10, we

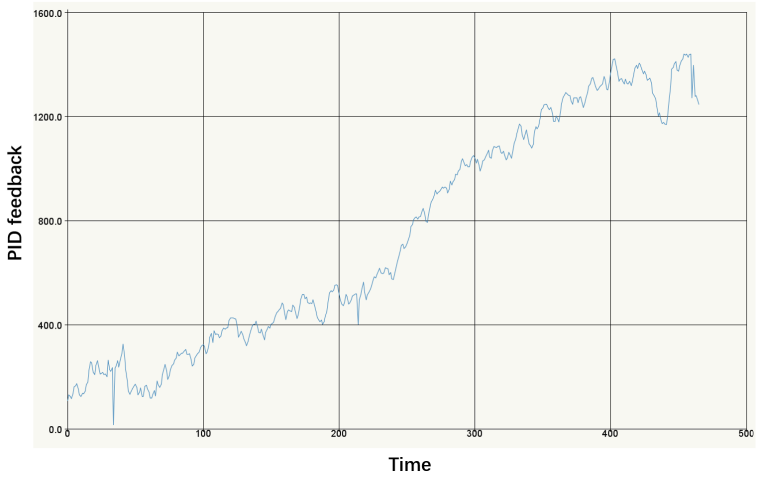


Figure 2.10: PID feedback when two robots fight each other

can see that the PID feedback value continues to rise rapidly, indicating that the robot is attempting to attain the required velocity but is unable to do so owing to a significant external force.

The box, on the other hand, could be pushed forward, resulting in smoother PID feedback which is shown in the graph 2.11. The graph 2.11's curve depicts the variation in PID feedback as the

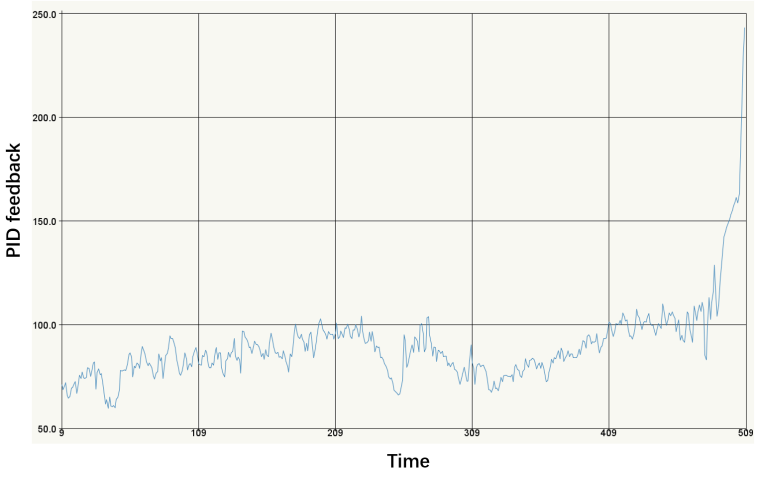


Figure 2.11: PID feedback when robot pushes a box

robot continued to push the box until it hit the wall. It can be seen that the PID feedback fluctuates around 80 in the stages of pushing the box and spikes once it hit the wall.

As for pushing a stationary object like a wall, this situation is similar to pushing a box because the external force applied to the robot is not as strong as the force when two robots fight. It's gentle and the PID feedback will soon reach a state of equilibrium as can be seen in graph 2.12. Graph

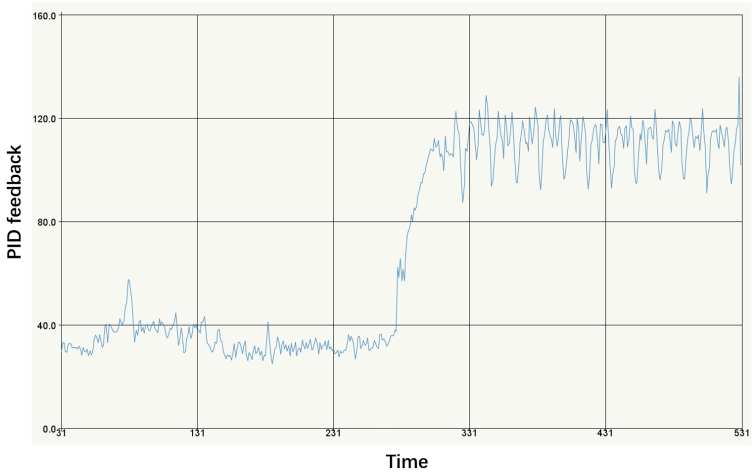


Figure 2.12: PID feedback when robot moves on a smooth surface and encounters a wall

2.12 illustrates the change in PID feedback when the robot moved from a smooth surface until hit a wall and attempted to push it. When the robot pushed the wall, the PID feedback was around 110, which is slightly higher than that when the robot pushed the box. This may be because the sliding friction force applied by the box on the robot is smaller than that applied by the wall.

To make robot distinguish the box from an oncoming robot, I set a threshold as 100. When the feedback is larger than 100, then the robot will believe it encounters a robot and need to do turning around movement, otherwise, it will prepare for pushing the box.

This detection of objects is effective when the robot collides with the front of another robot. If the robot 1 collides with robot 2 from robot 2's side or rear, the PID feedback will be similar to the ones when the robot collides with the box, because the bump sensors are on the robot 2's front and are not pressed, so it won't cause a head-to-head fight between two robots, which may lead the robot to be pushed out of the enclosed area. Graph 2.13 illustrate this point. This graph shows the PID feedback when a robot pushes another robot from its back. The value of the PID feedback floats around the threshold makes it difficult for object recognition. This problem is difficult to solve under the current conditions and the solution of this problem I will mention in section 5.2 future work.

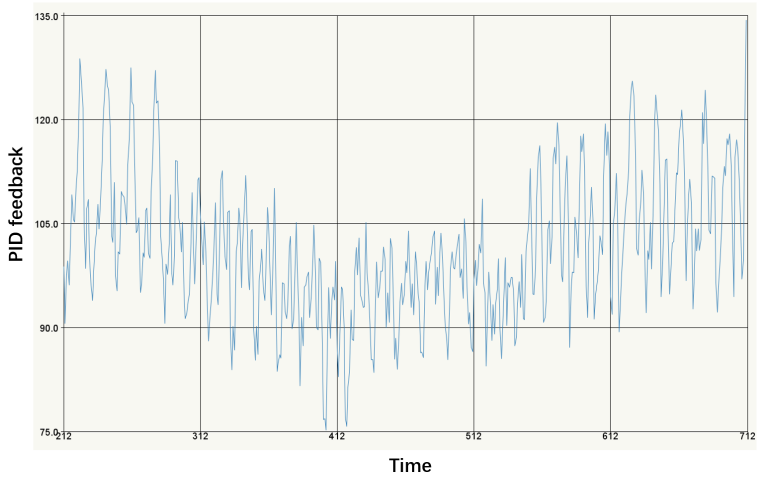


Figure 2.13: PID feedback when robot pushes another robot from its side

2.3.7 IR communication board

This project utilises a short range communication board designed by Dr. Paul O'Dowd from the University of Bristol. This board has been utilised without major modifications, and serves to increase the functionality of the basic Pololu 3Pi+ robot. This board is visible in figure 2.14. More details can be found online[32].

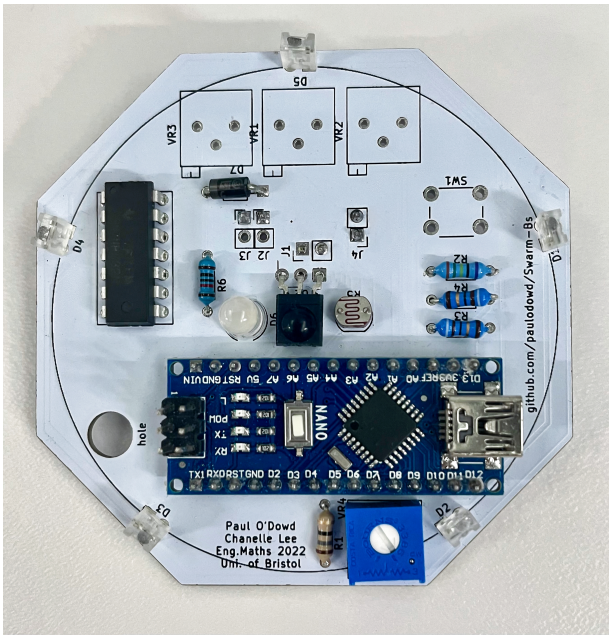


Figure 2.14: Communication board

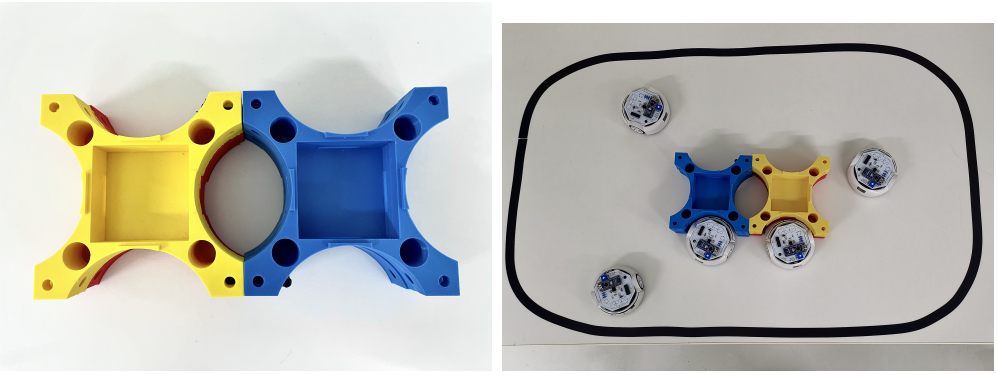
This communication board enables the Pololu 3Pi+ robot to conduct local infrared communications with a maximum range of approximately 30cm, sending or receiving a maximum character length of 28 bytes. Importantly, the board can only send and receive in discrete (independent) intervals, and the board is not capable of determining the location or direction of transmission of a message. For example, a robot will receive messages from all directions. Similarly, robots transmit messages in all directions around themselves (i.e. a broadcast). Due to the frequency of sending and receiving information, in practical applications, multiple robots sending information to each other may cause conflicts, resulting in some robots not receiving information immediately and delaying their actions. However, the board does have an error correction algorithm implemented, meaning that corrupted information is ignored, and only the efficiency of receiving messages is impacted. This will be discussed later in the chapter 3 and section 5.2.

2.3.8 Box Design and Considerations

The box for the box-pushing task is 3D printed which has four inwardly recessed circular sides(shown in figure 2.15a). Ideas for the shape and size of the box were developed by me and Dr. Paul O'Dowd and the model of the box was designed by Dr. Paul O'Dowd from the University of Bristol, and the design files are available in the appendix A. Each corner of the box is perforated with a hole so that it could be connected to another box and secured with a tether (for example, a zip-tie) to increase its size. The box's top has four protrusions and the bottom has four recesses, allowing the boxes to be stacked to enhance their height. There is also a square depression in the center of the box to reduce the weight of the box.

The box I used consists of four base boxes, as shown in the figure 2.15a, and it has two tiers, with two base boxes on each tier. The box's width and length are both 150mm, and its height is 30mm. It is important that two boxes are stacked up, which means that the communication between robots is obstructed. This is because the line of sight between the transmitter robot and a potential receiver robot is blocked. This prevents a robot from the opposite side, which is about to push the box in the opposite direction, from receiving the message. So the possibility of two robots pushing boxes from opposite sides of the box at the same time is reduced, which is a phenomenon that can lead to stagnation in the box-pushing task.

Because of the length of the total box, when only one robot is pushing the box, the robot may



(a) Image of the box (b) Image of the arena

Figure 2.15: Box for box-pushing task and experimental arena

push the box in a circle (because there is no robot to help push the other side). The trajectory is shown in the figure 2.16. As a result, the box cannot be pushed out of the field efficiently by a single robot, creating a task that is more difficult for a single robot. In contrast, when two robots are pushing at the same time, the box is more likely to be pushed along a straight trajectory and be pushed out of the field more quickly.

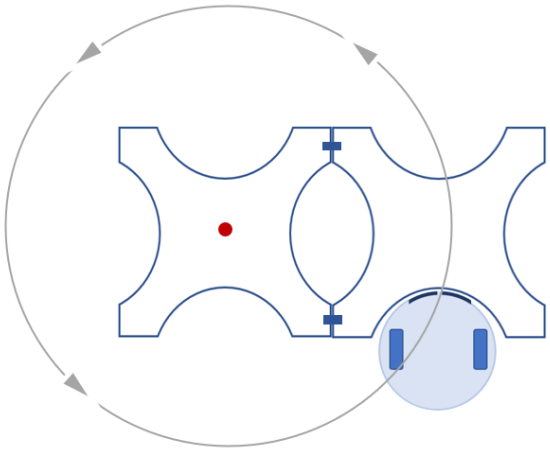


Figure 2.16: Trajectory of single robot pushing the box: take the red dot as the center of the circle and the distance between the centers of the two boxes as the radius

The box has been designed specifically with some semi-circle cut-outs to accommodate the robot. Using this design of box, the robot is better able to find the point of application of force as it pushes. The radius of the cutout is 55mm, which is slightly larger than the robot's radius.

The experimental arena is shown in the figure 2.15b. The experimental arena is a rectangle with a length of 95cm and a width of 55cm. The arena is enclosed by a black line set away from the edge, which the robots are able to detect and navigate away from, to prevent moving off the arena surface. This also allows robots to complete a box-pushing operation, assuming that the box has left the arena when the black tape is encountered.

2.3.9 System Parameters

Some specific parameters are also used in this experiment in order to make the performance of the system better. The values of these parameters are listed below:

- PID feedback threshold: The value of the threshold is 200, which means if the feedback is greater than 200, then the robot would know that it has bumped into another robot.
- Bump sensor threshold: The value of bump sensor threshold is 1600. The reading of bump sensor larger than 160 means that the robot has bumped into something.
- Line sensor threshold: The value of line sensor threshold is 1900. When the reading of line sensor is greater than 1900, the robot has detected the black line and is at the arena's edge, ready to turn around.
- Waiting time values: After the robot hit the box, the robot would wait trying to recruit other robot to help. Whether the robot recruits another robot or not, it will start pushing boxes after waiting 20s.
- Interval between sending and receiving messages: The robot transmits and receives signals every 100 and 150 milliseconds, respectively.
- The basic forward velocity is 0.45. This is an appropriate speed for a single robot to move the box.
- For the PID controller, K_p, K_i and K_d(the gain of P, I, D) for direction controller is 2.5, 0.0001, 0.0, for speed controller is 80.0, 0.1, 0.0.

2.4 Experiment Methodology

This section covers in detail the experiment's procedures and the data collected throughout the experiment. Concurrently, I pointed out the potential problems with the experiment and what I did to control them.

2.4.1 Experiment Variables

The independent variables in this experiment are

- Communication type: There are four communication methods: no communication(NC), implicit communication(IC), explicit communication-recruitment (EC-R), explicit communication-halting (EC-H).
- The number of robots is varied to 2, 3, 4, and 5 robots.

Therefore, there are four experimental scenarios conducted with the abbreviation *NC, IC, EC-R* and *EC-H*. These are each repeated with the numbers of robots 2,3,4 and 5. Every combination of scenario and number of robots is repeated for 5 trials. Therefore, this investigation has made 80 trials in total. For these experiments, I captured the **dependent variables** as shown in the following:

- Number of times the robot cooperated, $N_{cooperate}$: The times that more than one robot push the box together on the same side, indicating that the robots successfully assist each other.
- Number of times the robot pushed against each other, $N_{against}$: This means the times that two or more robots push on different sides of the box. Since the opposite forces cancel each other out, the box cannot be pushed.
- Number of times the robot pushed alone, N_{alone}: This means the times that just one single robot pushes the box and no one helps it.
- Number of times the robot got stuck, N_{stuck}: When several robots meet each other, it is likely that the robots may mistake each other for a box and continue to push each other, causing them to get stuck, section 2.4.2 illustrates this situation. If the robots continue being stuck for more than 30 seconds, then N_{stuck} will increase by one.

- Number of times the robot be pushed out of the area, N_{out}: When the robot is pushed by another robot from the its own side and back, it is more likely to be pushed out of the area enclosed by the black line, and this number is used to count the number of times the robot is pushed out.
- Number of times the robot waited, N_{wait}: This measurement only be collected when using explicit communication which counts the times that robots wait for help.
- Number of times the task completed, N_{task}: This means how many times the box is pushed out of the arena successfully.

The **independent variables** and **dependent variables** of the main experiments in this project are summarized in the table 2.1.

Table 2.1

Category	**Variable**	**Definition**
Independent variables	Communication type	There are four communication methods: no communication(NC), implicit communication(IC), explicit communication-recruitment (EC-R), explicit communication-halting (EC-H).
	The number of robots	The system size of the robot: From 2 robots to 5 robots.
Dependent variables	$N_{cooperate}$	Number of times the robot cooperated
	$N_{against}$	Number of times the robot
	N_{alone}	Number of times the robot pushed alone
	N_{stuck}	Number of times the robot got stuck
	N_{out}	Number of times the robot be pushed out of the area
	N_{wait}	Number of times the robot waited
	N_{task}	Number of times the task completed

With these variables described above, I wish to discover the inherent connection between the number of completed tasks and the different robot behaviors. Through these relationships, I can more easily identify the factors that contribute to the accomplishment of the task and the areas that need improvement.

In order to keep the measurements I collected are reliable, robust and meaningful, there are some necessary **controlled variables**:

- The duration of the experiment: The duration of every set of experiment is 5 minutes. If the duration is too short, the robot may not even have enough time to complete the task for once. If the duration is too long, then the task of collecting data becomes onerous.

- Ambient light: This factor influenced the experiment a lot. In this experiment, the ambient light should preferably be uniform, if the light(such a the sunlight) comes from just one side, then the line sensors of the robots would fail, the robots will find it hard to detect the black line and will decide to cross it.

- The size and shape of the experiment arena: The size of the arena should not be too small or it would be crowded and in chaos. If it is too large, then it would take a long time to see the box pushing happen or it would take a large number of robots. A well-sized arena can save both time and resources. The shape of the field should better be curved without sharp angles, as the robot will not be able to turn around in time when it hits a sharp corner.

- Box size and weight: The size of the box should be suitable for the robot. This has been introduced in section 2.3.8. And the box should be adjusted to a weight that enables at least two robots to push the box, as the minimum cooperative cell is two robots.

- The color of the experimental arena surface: The color of the surface should be light enough so that the robots could distinguish the black line. The surface color used in this experiment is pure white

- The weight of the robot: Although the robots in the scenario with no communication and implicit communication do not have to utilise the communication board, In order to maintain the same weight of the robot so as not to affect the speed of the robot, the robots in all experiments were equipped with communication boards.

2.4.2 Procedure

To make the experiment fair and to enable the experiment to proceed well, there are some human activities that need to be regulated. All robots are triggered sequentially and calibrated by pressing their bumpers' skirts. Once all robots have been activated, they will be evenly distributed around

the arena's edge before the box is placed in the centre point and then the clock starts to count for 5 minutes. So when the clock starts, every robot is start moving. Taking five robots as an example, the initial positions of the robots are shown in the figure 2.17.

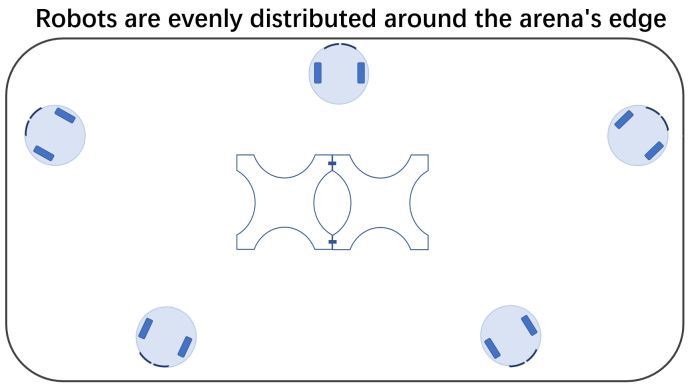


Figure 2.17: Initial positions of the robots, the robots are evenly distributed around the arena's edge

During the experiment, if the robots get stuck with the box or other robots, human will start to count for 30 seconds, if the robots are stuck for over 30 seconds, then N_{stuck} will increase by 1, after that, the robots will be separated manually and evenly distributed around the arena's edge. The process of dealing with the stuck robots is shown in the figure 2.18.

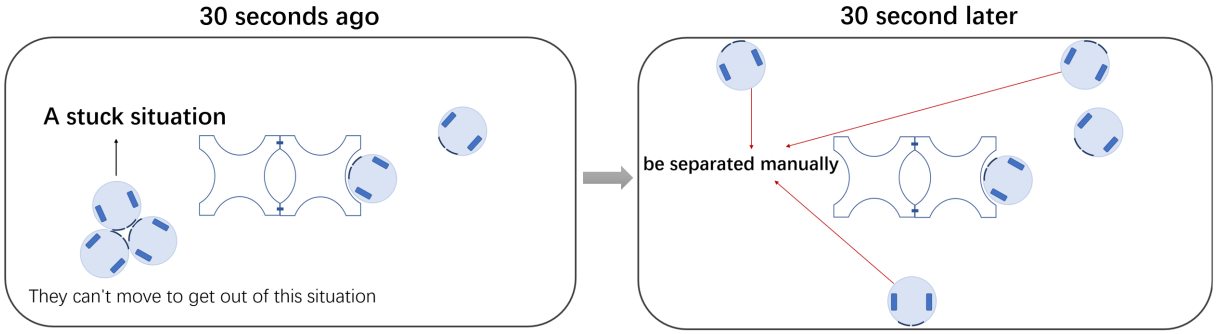


Figure 2.18: Process of dealing with the stuck robots: the robots will be separated manually after stuck for 30 seconds

If a robot is pushed out of the arena, then it will be placed on the edge closest to where it was pushed out, and N_{out} will increase by 1. The position where the robot is placed is shown in the figure 2.19.

If the box is pushed out, then it will be placed in the middle of the arena. This ensures that each pushing task is fair (every time the box is pushed from the middle of the arena to the outside). The

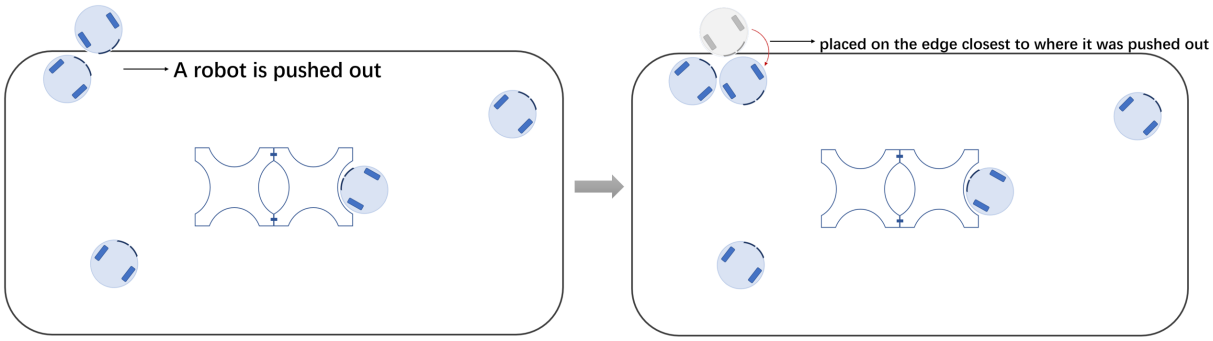


Figure 2.19: Process of dealing with the robot being pushed out: the robot will be placed on the edge closest to where it was pushed out

position where the box is placed is shown in the figure 2.20.

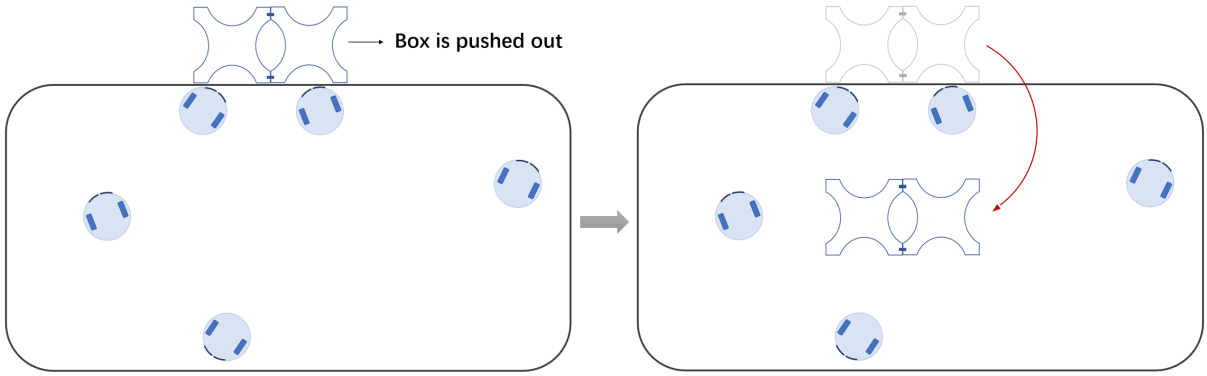


Figure 2.20: Process of dealing with the box being pushed out: the box will be placed in the middle of the arena

3 Results

To explore whether and which local communication can improve swarm box pushing, I collected data on the various behaviours of the robots in the experiment and compared the results in two ways: the type of communication method and the number of robots. These comparisons helped to find the best communication methods and to explore the scalability of the robotic system. This chapter proceeds as follows. First, the results of varying communication types are discussed in section 3.1. Then the results of varying number of robots are demonstrated in section 3.2. Finally, the conclusion and the issues need to be further investiget are summarised, before drawing final conclusions in chapter 5.

3.1 Varying Communication types

In this section, I compare all the experimental results according to various communication methods (*NC, IC, EC-R, EC-H*, as detailed in section 2.2.1), and the results are shown as follows.

The chart in the figure 3.1 shows the number of times the robot successfully completed the task with different communication methods. The variance and mean values show that implicit communication is the most effective. The use of implicit communication resulted in the highest and more consistent number of successful task completions on average. In contrast, explicit communication is less powerful, particularly EC-H. It shows that implicit communication works best in terms of task success rates alone.

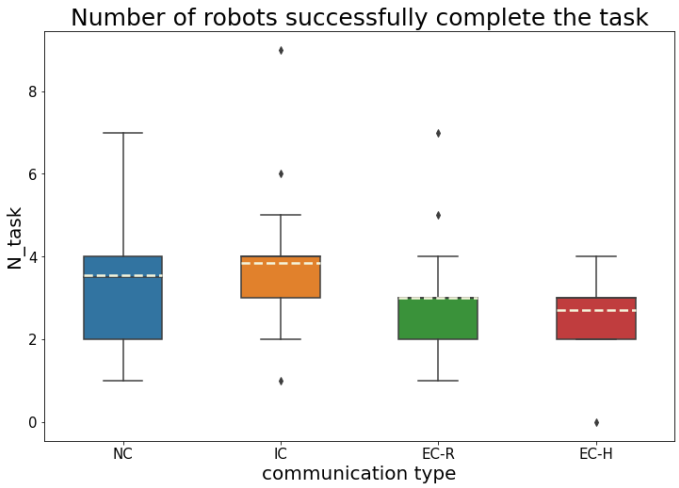
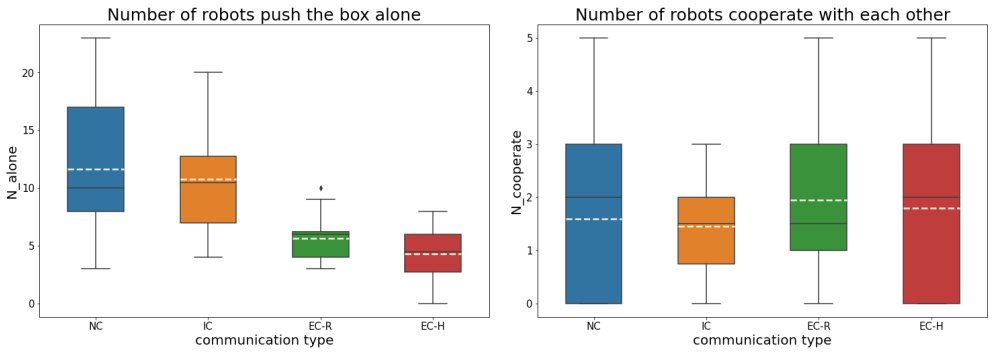


Figure 3.1: the number of times the robot successfully completed the task(note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation.)

It can be seen from 3.2a and 3.2b that using explicit communication can reduce the number of times the robot pushes the box alone and increase the number of instances of robot cooperation. This indicates that explicit communication did promote cooperation between robots. However, this increase in cooperation did not seem to be very helpful in task completion, which can be observed by a mean value of *N_task* in figure 3.1



(a) Number of robots push the box alone (b) Number of robots cooperate with each other

Figure 3.2: The number of robots push the box alone or collaboratively (note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation. The scale of the two coordinate systems' axes are not consistent))

Figure 3.3 shows that using implicit communication (IC) makes the robot more likely to push boxes from opposite directions with other robots, while explicit communication-Halting (EC-H) could help solve such problem because explicit communication-Halting is designed for avoiding this situation because if the robots with EC-H start to push the box, they will communicate with other robots to tell them just stay in place, so other robots won't block their way.

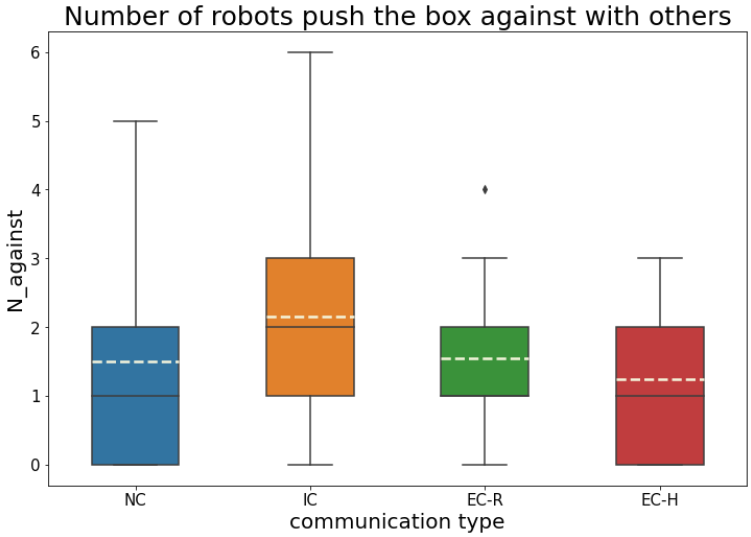
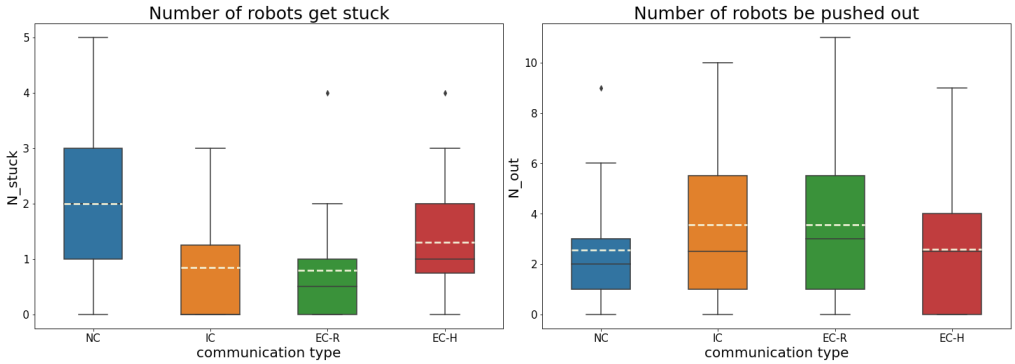


Figure 3.3: Number of robots push the box against with each other(note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation.)

Figure 3.4a shows that robots using Implicit Communication (IC) and Explicit Communication-Recruiting (EC-R) were less likely to get stuck. And 3.4b demonstrates that the number of robot be pushed out of the arena is the least with no communication.

For all the aspects mentioned above, the differences in numerical terms between systems with different communication methods are not notably significant. This is observable by the overlap of the error bars, meaning that the results were not particularly separable. From figure 3.5 we can see, the aspect of explicit communication that differs the most from the other communication methods is that both types of explicit communication have a certain amount of waiting time before the box be pushed.

I suppose the variable N_{wait} shown in figure 3.5 is the major element limiting the functioning of the explicit communication system. This is due to the fact that, while waiting for assistance, the



(a) Number of robots get stuck (b) Number of robots be pushed out

Figure 3.4: The number of robots get stuck and be pushed out (note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation. The scale of the two coordinate systems' axes are not consistent))

robot does nothing to assist other robots in locating a box or do any useful operations, other than continually sending signals to other robots informing them that it is waiting. Therefore, explicit communication appears to cause waiting behaviours and so a reduction of efficiency overall.

3.2 Varying Number of Robots

In order to evaluate the scalability of the robot system and to find the appropriate size for the robot system, I set up a set of experiments in which the number of robots varied from two to five when using different communication methods, detailed in section 2.3. Due to the limitations of the experimental arena's size, five robots seemed to make the arena a little crowded, so set maximum number no more than five should be a good choice.

From figure 3.6, which shows the sum of N_{task} when all communication types combined together, we can see that the robotic system performs the best when there are four robots in the system. When the size of the system gets larger, the effectiveness of the system began to decline.

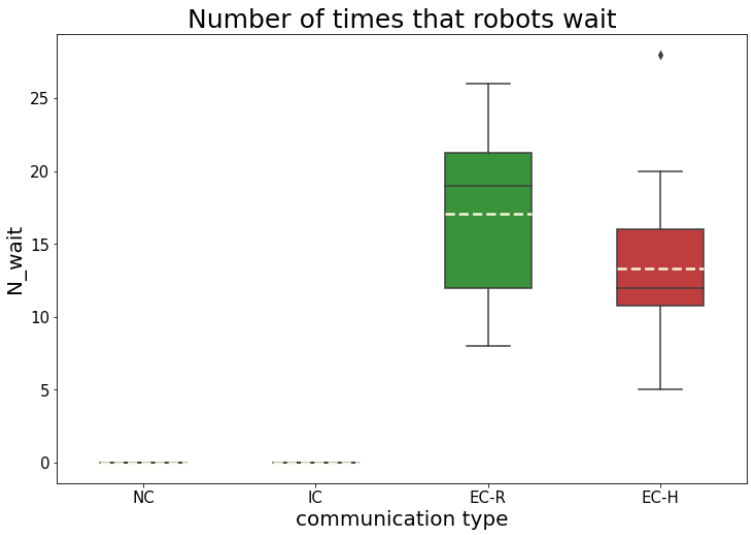


Figure 3.5: Number of times that robots wait(note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation.)

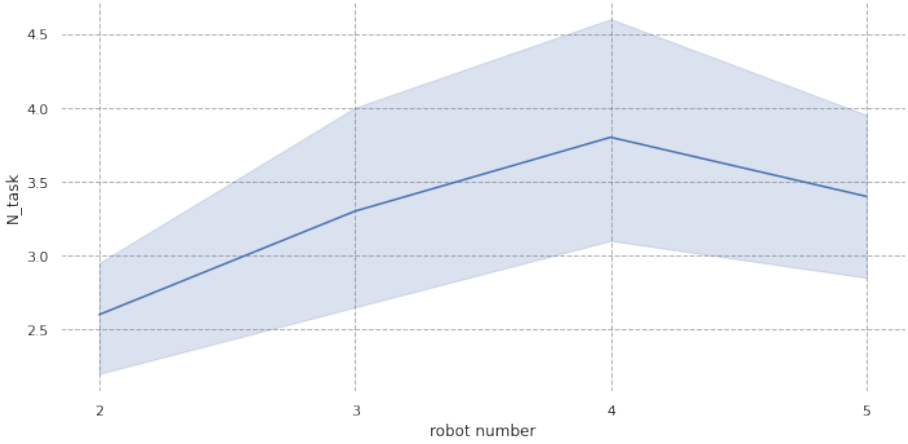


Figure 3.6: Result of N_{task} when all communication types combined

Figure 3.7 demonstrates the effectiveness of different sized robotic systems using different communication methods to complete tasks. No communication and implicit communication have similar trends, with both being most effective when there are four robots in the system, while explicit communication is less effective except when the robot system has just two robots. The performance of systems using explicit communication does not vary much as the number of robots rises, and as the number of robots increases, the performance gap between explicit and the other two communication techniques narrows. This indicates that crowding in the arena has less effect on systems

with explicit communication as opposed to systems with implicit communication. It might due to the reason that the explicit communication makes the collective behavior of robots more orderly, but the cost is that it takes longer to achieve this performance.

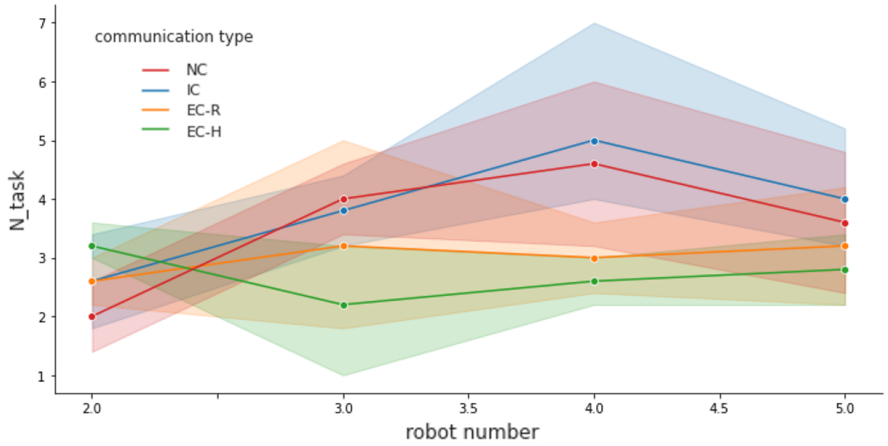


Figure 3.7: Results of varying numbers and communication types of robots

3.3 Summary

As I initially suspected, adding explicit communication does increase the number of times the robots work together, but it also significantly reduces the number of times the robots push the box alone. This is likely due to the fact that when using explicit communication, the robots spend a certain amount of time waiting in place before pushing the box, and this time is not utilized very efficiently. In addition, not every waiting period is for box pushing; occasionally the robots encounter another robot instead of a box, which renders the lengthy waiting period pointless.

And when using EC-H, the issue of time waiting got worse. When there is one or more robots pushing the box, then, other robots will stop moving so as not to obstruct the path of the robot pushing the box. While this does have some effect as $N_{against}$ and N_{out} are reduced, the number of times the robot gets stuck increases and the number of times it pushes the box decreases due to less movement. This makes it less probable that robots using this strategy would accomplish the task. So I stopped improving EC-H and shortened the waiting time of EC-R from 20 to 10 seconds and carried out a series of experiments, which are detailed in the section 4.2.

When a single robot pushes a box, it can only press one side of the box and move in an approximate circular trajectory so it can hardly push the box out of the black line in a single motion. However, two robots operating in tandem can push both sides of the box simultaneously makes the pushing more efficient. Nevertheless, it seems that a single robot repeatedly pressing the box in various directions could also accomplish the task. Therefore, the robot performed rather well while pushing boxes with just Brownian motion although there is no sensor to guide its movement, and no explicit mechanisms to facilitate cooperation. To explore this question more deeply, I conducted additional experiments described in the next chapter, section 4.1.

Figure 3.6shows that the system performance decrease with the increase number of robots. The reason for this can be observed during the experiments, the greater the number of robots, the higher the likelihood that they will get entangled since they would collide with one another more often. However, I also observed that as the number of robots increased, the frequency with which they received the message decreased (when a robot receives a message, the communication board will flash green, otherwise red). To investigate whether robots interfere with each other's message conveyance, I conducted a series of tests (described in the next chapter, section 4.3) to investigate whether the probability of receiving a message varies with the number of robots.

4 Follow-up Experiments

Chapter 3 discussed several unexplained phenomenon in the summary (section 3.3). For this reason, this chapter explores these phenomena further with a set of additional experiments. The additional experiments are single robot box pushing in section 4.1, using explicit communication-recruitment with 10s waiting time for box-pushing task in section 4.2, and message sending and receiving test in section 4.3.

4.1 Additional experiment 1: Single robot box pushing experiment

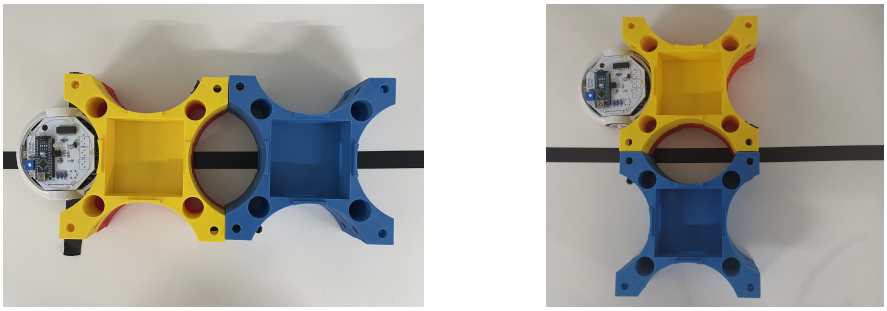
Since both the Brownian-only and implicitly communicative robotic systems relied on a higher number of individual box pushes to complete the task, in order to investigate the reason why the NC system worked well, I am going to add a set of experiments in which the robots are made to push boxes individually.

4.1.1 Experiment method

There are two groups, one group will use two bump sensors to sense the position of the box and equipped with PID to adjust the forward direction, and another group will not use any sensors and the direction of heading is changed randomly every 0.5 just like Brownian motion mentioned in section 2.2.1.

Each experiment will be conducted for 10 seconds and the coordinates of the box will be recorded. Both groups of robots will push the box in vertical and lateral positions as shown in the figure 4.1 which may occur in the box-pushing task, each position will be used for 10 experiments.

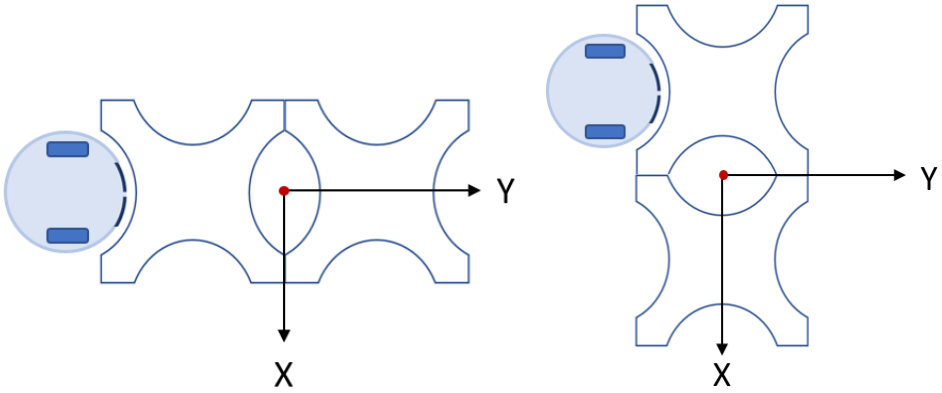
And the variables I am going to measure is the offset of the box from its initial position after



(a) The robot pushes the box lengthwise (b) The robot pushes the box laterally

Figure 4.1: Different positions of the robot pushing the box

being pushed by the robot for 10 seconds. The establishment of the coordinate system is shown in the figure 4.2.



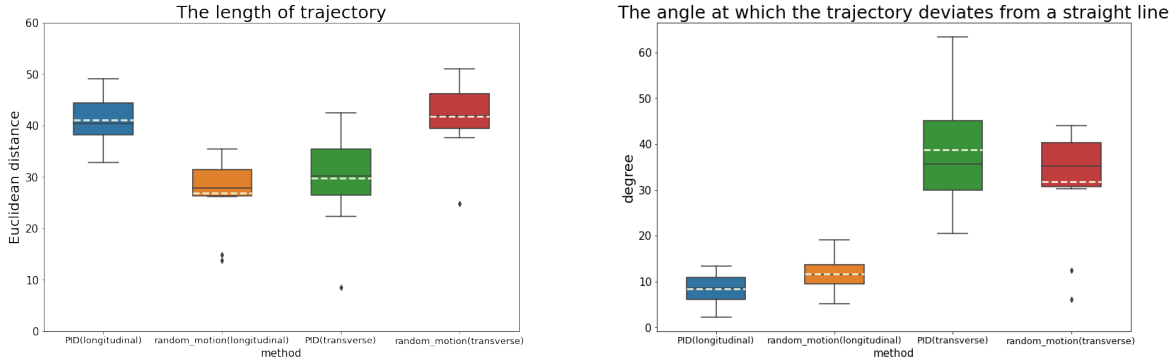
(a) Coordinate system when the robot pushes the box lengthwise (b) Coordinate system when the robot pushes the box laterally

Figure 4.2: Coordinate systems

4.1.2 Result

A straight trajectory for the boxes is the best result, because pushing the boxes in a straight line can lead to the shortest trajectory and the shortest time spent when the robot is pushing the boxes. So in this experiment the distance of the box from the origin and the angle of deviation of the box from the y-axis were measured, to investigate which method of pushing the box would result in the box travelling a longer distance in a fixed amount of time with a smaller angle of deviation. The results

are shown in the figure 4.3.



(a) The length of trajectory

(b) The angle of which the trajectory deviates from a straight line

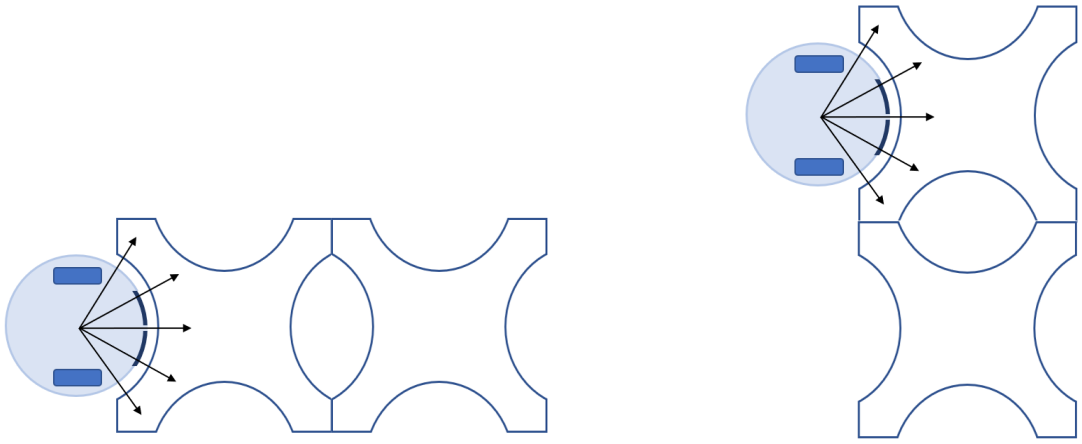
Figure 4.3: Different positions of the robot pushing the box(note:the beige dashed line is the mean, the solid black line is the median, and the dots are outliers. The error bars are the standard deviation.)

Observing the results for longitudinal box pushing alone in figure 4.3, the mean of trajectory and deviated angle shows that the PID-equipped robot pushed the box further and with less trajectory deviation in the 10 seconds experiments, as anticipated. In contrast, when the box is pushed in a laterally direction, the robot in Brownian motion performs substantially better in terms of trajectory length and deviation. Based on my observations during the experiment, I suppose that this is related to the way the robot applies force when pushing the box.

The PID equipped robot will use bump sensors to sense the posture of the object. If the object is on the left of the robot, the robot's right wheel will speed up so that the robot's tangent is as parallel as possible to the tangent of the box, using this method of pushing the box, it is very effective for the robot to push the object at the centre line of the object. However, if the robot is not at the centre of the object, the robot may push the object in a circle with the other end of the object as the origin.

If the robot in Brownian motion is pushing a box at the centre of the box, the robot has no logical direction of travel and can easily push the box to the side, causing it to leave the robot. However, if the robot is not at the centre line of the box, then the random direction of the robot could potentially help the box to move further. As shown in the figure 4.4b, the robot's forward direction, if pointed towards the centre of the box, achieves a delicate equilibrium, and although

the box's trajectory may deviate at an angle, it does not separate from the robot, allowing the robot to keep pushing the box.



(a) The robot in Brownian motion pushes the box lengthwise

(b) The robot in Brownian motion pushes the box laterally

Figure 4.4: Different positions of the robot in Brownian motion pushing the box

4.2 Additional experiment 2: Using explicit communication-recruitment with 10s waiting time

4.2.1 Method

The details of this robotic system is introduced in chapter 2. And here in this experiment, I changed the parameter of waiting time from 20 to 10 seconds, expecting better performance with less waiting time waste. The result of this experiment is shown in the figure 4.5.

4.2.2 Result

Figure 4.5 demonstrates that the EC-R system with the shorter waiting time outperforms the other explicit communication communication methods, tying for first place with the system without explicit communication when there are three robots. Nonetheless, as the number of robots rises, its advantage diminishes and its performance starts to deteriorate. Due to the lower waiting time of

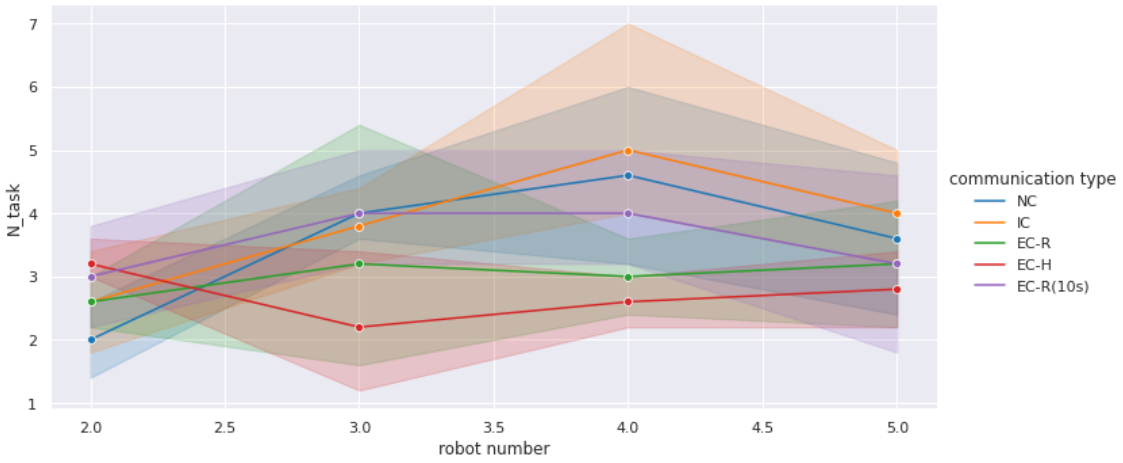


Figure 4.5: N_{task} of EC-R(10s) compares with other communication methods

this system, its explicit communication features are not as evident, and as the number of robots rises, the performance of this system tends to be similar to non-explicit communication.

The detailed results of EC-R(10s) is shown in the figure 4.6, here we can find some relationships between different behaviours and the number of times that robots complete the task.

In figures4.6(b) and (c), the averages of N_{alone} and $N_{cooperate}$ fall between the values of explicit and non-explicit communication, indicating that the impact of explicit communication diminished. Notable is the significant decrease in N_{stuck} compared to the other communication methods, but the concurrent increase in the number of robots being pushed out of the arena, possibly because more robots were pushed out of the arena during mutual confrontations as opposed to sustained confrontations. Due to the reduction in waiting time, robots may have had more time to explore and seek for the box, resulting in an increase in N_{wait}. All of these parameters enhanced EC-R(10s)'s performance but it is still slightly inferior than IC.

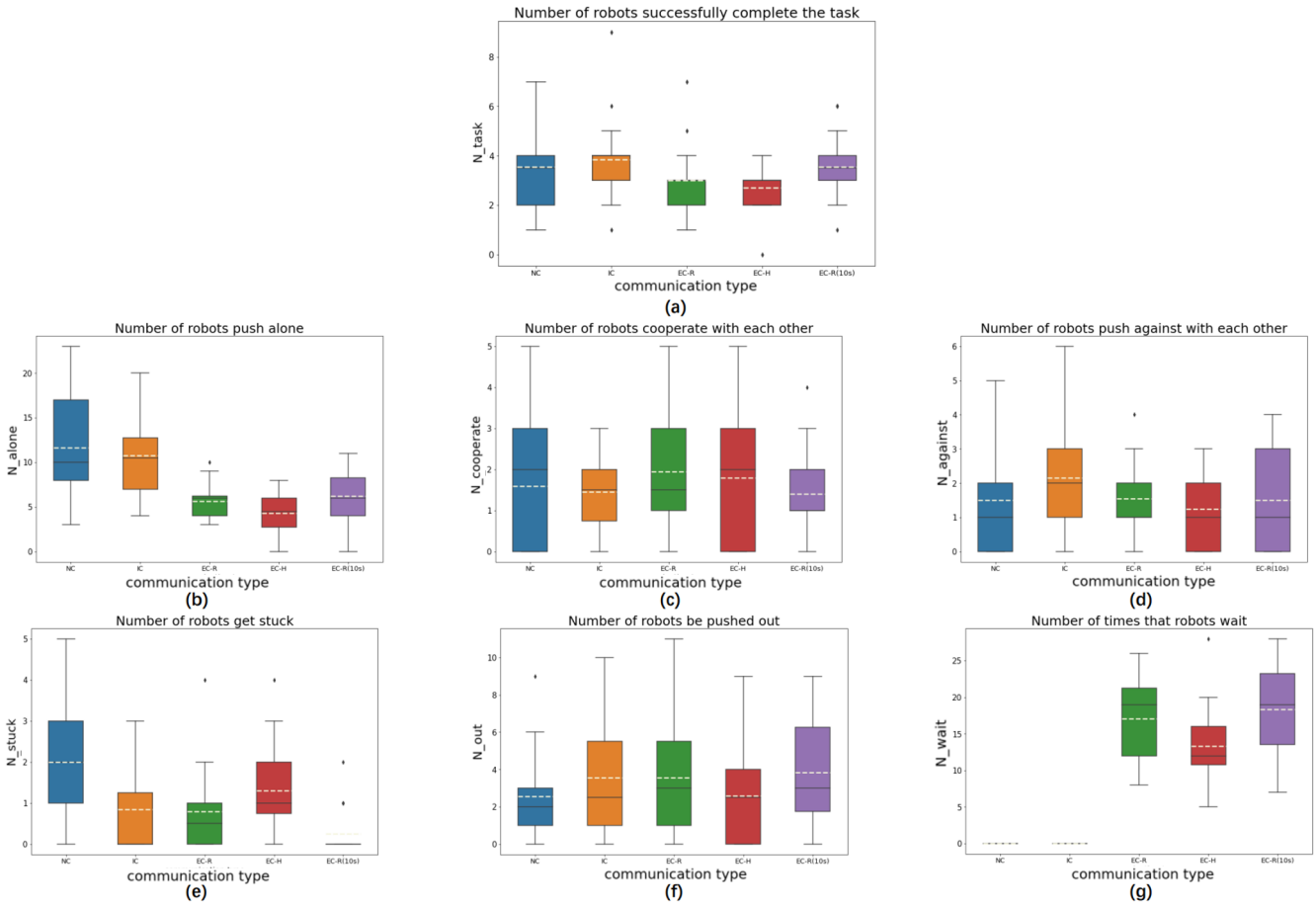


Figure 4.6: N_{task} of EC-R(10s) compares with other communication methods(note: The scale of the seven coordinate systems' axes are not consistent)

4.3 Additional experiment 3: Message sending and receiving test

4.3.1 Method

This experiment is designed to test and evaluate the reliability and efficiency of explicit communication under more controlled circumstances. In this experiment, I will make every robot communicate a message with each other five hundred times using communication board, and the quantity of messages successfully received by robots will be recorded. The distance between each adjacent robot is 15cm. I chose 15cm because when the robots work together to push the box, they are about

15 centimeters apart. The placement position of the robots is shown in the figure 4.7. Under these experiment conditions, the robots do not move or attempt to complete the box pushing task. To ensure fairness, each experiment will be repeated ten times, and each robot will be used. There are five robots in total, which means each robot will be used to collect data twice. To make the evaluation of the communication reliability and efficiency, the total number of robots used is varied. The hypothesis is that the number of robots is having an impact on the stability of communication due to message signal interference.

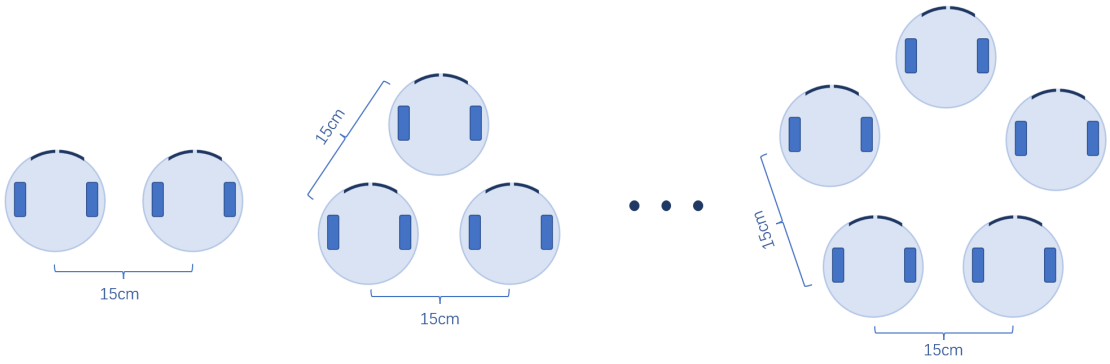


Figure 4.7: The placement position of the robots

4.3.2 Result

The collected result is shown in the figure 4.8.

The result from figure 4.8 shows that as the number of robots rises, the transmission and reception of communications between robots will be impacted. Considering that each robot transmits 500 messages to the outside world, the total number of messages received might be expected to grow as the number of robots increases. However, the robots experience interference in receiving the messages, resulting in fewer messages received by each individual robot.

From figure 4.8, it is possible to observe that in an ideal situation with two robots less than 500 messages are received. With just two robots, message interference is not expected. However, due to the design of the communication board (see section 2.14), a robot can only send and receive in independent periods. So this loss of messages received may be due to a lack of synchronisation between the sending and receiving robots. However, taking the situation with two robots as ideal, figure 4.8 shows that the performance of communication degrades significantly with the addition

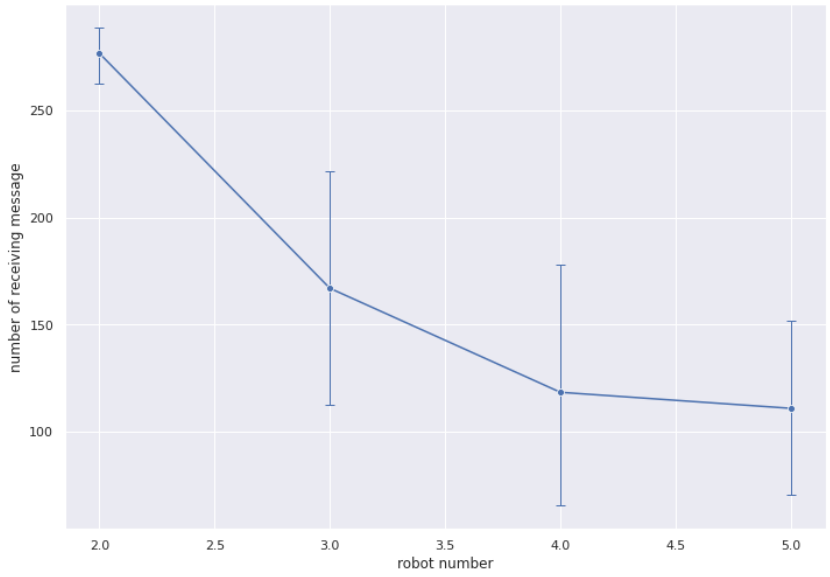


Figure 4.8: Number of received messages

of more robots, rather than staying consistently poor. Furthermore, the error bars broaden after two robots, indicating that the performance is more variable with the addition of further robots.

As the ability of the robots to receive messages is worse at a number of robots of 4 than at a number of 3, this explains the increase in the gap between explicit and implicit communication when the number of robots is 4 in the figure 3.7, i.e. the robots utilising explicit communication spent time waiting for help from other robots, and further delay was caused by the robots not receiving messages in a timely manner.

4.4 Summary

In general, despite the improvement of explicit communication's waiting time, implicit communication continues to have the most impact. The following conclusions may be drawn from the experiment: First, the robot system's performance is proportional to the number of robots. If the density of robots is too low, it will take longer to locate the box, but if it is too high, it will create congestion and make moving the box more difficult. Therefore, it is crucial to choose an appropriate size for the robot system. In these experiments, the better number of robots was 4 in total. However, this must be understood to be relative to the size of the task environment, and how much space these robots therefore occupy, or their probability of collision. Therefore, a general recom-

mendation on this performance issue cannot yet be made.

Second, the posture of the robot pushing the box can also be learned from the robot that only does Brownian motion. The PID may be used to estimate the box's location more precisely in order to change the direction in which the robot applies force to the box. The use of explicit communication may enhance cooperation between robots to some degree, but it is required to establish an adequate waiting period so as not to postpone the robot's entry into the next round of box exploration, as well as to improve the immediacy of communication.

In terms of the complexity of hardware and software, implicit communication is simpler and more effective at accomplishing tasks than explicit communication. If improvements of the communication board are possible, explicit communication may also be enhanced and would be the subject of future work.

5 Conclusion and future work

5.1 Conclusion

This project has investigated a box-pushing scenario within swarm robotics. In the box-pushing task, swarm robotic systems with different communication strategies tried to push the box out of the black line. The systems with different local communication methods are compared with baseline(no communication).

The project had the following hypotheses initially presented in section 1.2.3:

1. My initial hypothesis on the aforementioned research questions is that local communication can really help swarm robots cooperate with each other and thus improve task performance, and that explicit communication is likely to be more effective because the messages conveyed by explicit communication are more unambiguous.

2. Regarding the factor that is most important to the task's success rate, I hypothesise it may be the number of times that robots successfully collaborate, since this should considerably increase the speed of pushing the box and the straightness of the trajectory when pushing the box.

3. I hypothesise that explicit communication may facilitate the formation of collaborative behaviors in a timely manner, and therefore be more efficiently overall. For example, an explicit communication can travel distance between robots nearly instantaneously, whilst implicit communication requires robots to first encounter one another physically.

These hypotheses can now be discussed in light of the results presented in chapter 3 and 4.

First, hypothesis 1 concerned the effect of local communication. In the result of the main experiment in chapter 3, the performance of implicit communication(IC) is better than the baseline but explicit communication is not as good as I expected, and worse than the baseline. After analyzing

the behaviors of robots with diffident communication methods during the experiment, I improved the system with explicit communication(EC-R) for its drawback, result is shown in the section 4.2. This result showed that EC-R(10s) which is the upgraded version of EC-R performed much better than the original one and the baseline, but still didn't catch up with IC.

So far, I can say that half of my hypotheses 1 is confirmed (local communication is effective), while the other half is less certain (explicit communication is superior) because, if the communication board is further improved(my ideas for further improvement of the communication board are shown in section 5.2), I anticipate that the benefits of explicit communication will become increasingly apparent.

Hypothesis 2 concerned the most important factor to the task success. This hypothesis is refuted as shown in the result of the main experiment(seen in chapter 3). Explicit communication lags behind the other two forms of communication, and the behavior that most distinguishes the EC system from the other systems is the waiting before pushing the box, whereas the robots in the other systems start pushing immediately after encountering the box, which saves a great amount of time; therefore, I believe the next improvement to EC will begin with increasing the time utilisation, so that the robots can complete more tasks within the time limit. The advantages of collaborative behaviour should come to the fore when the time spent on facilitating the formation of collaborative behaviour is saved.

Hypothesis 3 was supported to some extent. Because in the results of the main experiment, the systems with explicit communication did show more cooperative behavior than the other systems. But their cooperation has not been so timely. It is true that the EC can transmit messages through space, but since the communication board cannot provide its own location information, the robot receiving the message has no effective way to find the sender of the message, so it has to walk around the arena to search for boxes. It takes a lot of time to induce cooperative behavior.

After considering the findings from these hypotheses, the following general insights and recommendations can be made:

- Currently, implicit communication is the most effective and also the simplest.
- Realise the function of localization and navigation be of considerable assistance to systems with explicit communication.

- Improving the immediacy of collaboration prompted by explicit communication is needed.

This project has fulfilled the aims initially stated in section 1.2.2 and completed the objectives as stated in section 1.2.2 in the following ways:

- I had a thorough grasp of the topic of Swarm Robotics, in which this project is based, as a result of my extensive reading at the beginning of the project.

- After analyzing the current literature on swarm robots collaborating to push boxes, I selected a question that many researchers have raised but not clearly addressed as a research goal and built a task scenario for the study based on the available robotic resources.

- Once the experimental task was determined, I programmed the robot from the most fundamental action of pushing a box to the final, complete robotic system for seven weeks. During this period, a significant problem was resolved that proved to be very beneficial for robotic systems using implicit communication and PID feedback for object discrimination.

- While programming the robot, I set up the experimental arena and discussed with Dr. Paul O'Dowd to design a suitable box for the robot.

- I have designed four different robotic systems and compared them to one another. To conduct a thorough analysis of the trials, I established a large number of variables and collected data for discussion. Follow-up experiments were also added to clarify some of the issues that the experiment demonstrated.

- I have conducted a comprehensive evaluation of the performance of the four robotic systems, verifying and disputing my hypothesis, highlighting the benefits and drawbacks of systems using various communication techniques, and proposing solutions.

In this project, implicit and explicit local communication is deployed on a number of two-wheeled mobile robots to do the task of pushing boxes. These communications allow the robot to sense the existence of boxes, detect the types of objects it encounters, communicate its current state information with other robots, and perform box-pushing behaviors. The swarm robotics systems have good scalability due to its characteristics, however when the experimental environment changes, there is an ideal size for the robot system.

Overall, implicit communication is the best choice when considering the additional complexity of the system and the added value of using the communication board. Nonetheless, when explicit communication with updated parameters was introduced in additional experiment 2, the performance of explicit communication was considerably enhanced, indicating that there is still much space for improvement in explicit communication.

5.2 Future work

The improvement of communication boards will be a significant portion of future work, and communication boards with increased functionality will increase efficiency. The first issue that must be solved is the issue of several robots' communications interfering with one another. So that communications are not lost, the frequency of sending and receiving messages in the communication board must be further adjusted so that the board gets messages at a time that substantially overlaps the time at which they were sent.

The second function that could be added to the communication board would be to attempt to determine the direction of origin of the received messages in order to achieve the desired communication; the signal receivers are evenly spaced around the board's perimeter and could be polled to determine the direction.

Additionally, the upgraded communication board might contribute more to robot collaboration. When two robots are simultaneously pushing a box, they may detect the direction of each other by identifying the source of the message. This enables them to communicate the values given by their bump sensors during the push, which may improve their collaboration.

When a robot is identifying the type of object it is pushing, it is difficult to distinguish the robot from the box based solely on PID input when pushing from the side or rear of another robot. Using the communication board, if a robot detects that a box is in front of it and also gets a message from its front, it is probably pushing another robot (as the box blocks the signal when the robot is pushing and thus it could not receive any signal).

The above is a hardware upgrade proposal. On the software side, I believe it would be advantageous to simplify the system by combining some of the finite state machine's states. For example, the state of a robot recruiting other robots for assistance could be combined into the BoxPushing

state to save waiting time and simplify the process of box-pushing task. In terms of communication content, I believe that goal communication would be more beneficial than state communication, as stated by Balch[17], and in future work based on an improved communication board with positioning capabilities, it is anticipated that communication that includes both state and goal content will be a significant improvement to explicit communication.

A Appendix

```
//code for box design
$fn=100;

difference() {
    union() {
cube([150,150,30],center=true);
        rotate([0,0,90])translate([0,40,14]) cube([30,4,10], center=true);
        rotate([0,0,0])translate([0,40,14]) cube([30,4,10], center=true);
rotate([0,0,180])translate([0,40,14]) cube([30,4,10], center=true);
rotate([0,0,270])translate([0,40,14]) cube([30,4,10], center=true);



    }



// stacking recesses
rotate([0,0,0]
) translate([0,39,-11]) cube([32,6,10], center=true);
rotate([0,0,90]
) translate([0,39,-11]) cube([32,6,10], center=true);
rotate([0,0,180]
) translate([0,39,-11]) cube([32,6,10], center=true);
rotate([0,0,270]
) translate([0,39,-11]) cube([32,6,10], center=true);


// robot meeting points
translate([95,0,0]) cylinder(h=200, r=55,center=true);
translate([-95,0,0]) cylinder(h=200, r=55,center=true);
```

```
translate([0,95,0]) cylinder(h=200, r=55,center=true);
translate([0,-95,0]) cylinder(h=200, r=55,center=true);

translate([0,0,15]) cube([71,71,55],center=true);

    translate([0,65,0])rotate([0,90,0]) cylinder(h=1000, r=4, center=true);
translate([0,-65,0])rotate([0,90,0]) cylinder(h=1000, r=4, center=true);

translate([65,0,0])rotate([90,0,0]) cylinder(h=1000, r=4, center=true);
translate([-65,0,0])rotate([90,0,0]) cylinder(h=1000, r=4, center=true);

translate([-65,65,0])rotate([0,0,90]) cylinder(h=1000, r=4, center=true);
translate([-65,-65,0])rotate([0,0,90]) cylinder(h=1000, r=4, center=true);

translate([65,65,0])rotate([0,0,90]) cylinder(h=1000, r=4, center=true);
translate([65,-65,0])rotate([0,0,90]) cylinder(h=1000, r=4, center=true);


// Getting rid of some bulk
rotate([0,0,0]) translate([45,-45,0])rotate([0,0,0])
cylinder(h=1000, r=10, center=true);
rotate([0,0,90]) translate([45,-45,0])rotate([0,0,0])
cylinder(h=1000, r=10, center=true);
rotate([0,0,180]) translate([45,-45,0])rotate([0,0,0])
cylinder(h=1000, r=10, center=true);
rotate([0,0,270]) translate([45,-45,0])rotate([0,0,0])
cylinder(h=1000, r=10, center=true);


}
```

References

[1] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, Swarm robotic behaviors and current applications, *Frontiers in Robotics and AI*, vol. 7 2020, p. 36, 2020.

[2] C. Anthony, *3 ways robots affect the economy*, [EB/OL], https://www.investopedia.com/articles/markets-economy/091316/3-ways-robots-affect-economy.asp, Jul. 2021.

[3] E. Sahin, "Swarm robotics: From sources of inspiration to domains of application.," Jan. 2004, pp. 10–20.

[4] S. Camazine, J. Deneubourg, N. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*. English. United States: Princeton University Press, 2001.

[5] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, Swarm robotic behaviors and current applications, *Frontiers in Robotics and AI* [online], vol. 7 2020, 2020, ISSN: 2296-9144. DOI: 10.3389/frobt.2020.00036. available from: https://www.frontiersin.org/article/10.3389/frobt.2020.00036.

[6] G. Beni, "From swarm intelligence to swarm robotics," in *Swarm Robotics*, E. Şahin and W. M. Spears, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1–9, ISBN: 978-3-540-30552-1.

[7] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield, B. J. Nelson, B. Scassellati, M. Taddeo, R. Taylor, M. Veloso, Z. L. Wang, and R. Wood, The grand challenges of science robotics, *Science Robotics* [online], vol. 3, no. 14 2018, eaar7650, 2018. DOI: 10.1126/scirobotics.aar7650. eprint: https://www.science.org/doi/pdf/10.1126/scirobotics.aar7650. available from: https://www.science.org/doi/abs/10.1126/scirobotics.aar7650.

[8] S. Wolfram, Cellular automata as models of complexity, *Nature*, vol. 311, no. 5985 1984, pp. 419–424, 1984.

[9] J. Harvey, The blessing and curse of emergence in swarm intelligence systems, in *Foundations of Trusted Autonomy*, H. A. Abbass, J. Scholz, and D. J. Reid, Eds. Cham: Springer International Publishing, 2018, pp. 117–124, ISBN: 978-3-319-64816-3. DOI: 10.1007/978-3-319-64816-3_6. available from: https://doi.org/10.1007/978-3-319-64816-3_6.

[10] M. K. Heinrich, M. Divband Soorati, T. Kaiser, M. Wahby, and H. Hamann, Swarm robotics: Robustness, scalability, and self-x features in industrial applications, *it - Information Technology* [online], vol. 61 Oct. 2019, Oct. 2019. DOI: 10.1515/itit-2019-0003.

[11] T. Sugawara, K. Fukuda, T. Hirotsu, and S. Kurihara, "Adaptive probabilistic task allocation in large-scale multi-agent systems and its evaluation," in *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, 2010, pp. 1311–1312.

[12] Y. Tan and Z.-y. Zheng, Research advance in swarm robotics, *Defence Technology* [online], vol. 9, no. 1 2013, pp. 18–39, 2013, ISSN: 2214-9147. DOI: https://doi.org/10.1016/j.dt.2013.03.001. available from: https://www.sciencedirect.com/science/article/pii/S221491471300024X.

[13] C. Kube and E. Bonabeau, Cooperative transport by ants and robots, *Robotics and Autonomous Systems* [online], vol. 30, no. 1 2000, pp. 85–101, 2000, ISSN: 0921-8890. DOI: https://doi.org/10.1016/S0921-8890(99)00066-4. available from: https://www.sciencedirect.com/science/article/pii/S0921889099000664.

[14] Y. U. Cao, A. B. Kahng, and A. S. Fukunaga, Cooperative mobile robotics: Antecedents and directions, in *Robot colonies*, Springer, 1997, pp. 7–27.

[15] O. Holland and C. Melhuish, Stigmergy, Self-Organization, and Sorting in Collective Robotics, *Artificial Life* [online], vol. 5, no. 2 Apr. 1999, pp. 173–202, Apr. 1999. DOI: 10.1162/106454699568737. eprint: https://direct.mit.edu/artl/article-pdf/5/2/173/1661673/106454699568737.pdf. available from: https://doi.org/10.1162/106454699568737.

[16] E. Tuci, M. H. Alkilabi, and O. Akanyeti, Cooperative object transport in multi-robot systems: A review of the state-of-the-art, *Frontiers in Robotics and AI*, vol. 5 2018, p. 59, 2018.

[17] T. Balch, "Communication, diversity and learning: Cornerstones of swarm behavior," in *International workshop on swarm robotics*, Springer, 2004, pp. 21–30.

[18] N. White, J. Harwell, and M. Gini, Socially inspired communication in swarm robotics, *arXiv preprint arXiv:1906.01108* 2019, 2019.

[19] N. R. Hoff, A. Sagoff, R. J. Wood, and R. Nagpal, "Two foraging algorithms for robot swarms using only local communication," in *2010 IEEE International Conference on Robotics and Biomimetics*, IEEE, 2010, pp. 123–130.

[20] S. Torabi, "Collective transportation of objects by a swarm of robots," 2015.

[21] N. Ahmad and A. Agah, Plan and intent recognition in a multi-agent system for collective box pushing, *Journal of Intelligent Systems*, vol. 23, no. 1 2014, pp. 95–108, 2014.

[22] S. Li, R. Kong, and Y. Guo, Cooperative distributed source seeking by multiple robots: Algorithms and experiments, *IEEE/ASME Transactions on mechatronics*, vol. 19, no. 6 2014, pp. 1810–1820, 2014.

[23] B. Gerkey and M. Mataric, Sold!: Auction methods for multirobot coordination, *IEEE Transactions on Robotics and Automation* [online], vol. 18, no. 5 2002, pp. 758–768, 2002. DOI: 10.1109/TRA.2002.803462.

[24] A. Ghosh, A. Ghosh, A. Konar, and R. Janarthanan, "Multi-robot cooperative box-pushing problem using multi-objective particle swarm optimization technique," in *2012 World Congress on Information and Communication Technologies*, IEEE, 2012, pp. 272–277.

[25] M. J. Mataric, M. Nilsson, and K. T. Simsarin, "Cooperative multi-robot box-pushing," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, IEEE, vol. 3, 1995, pp. 556–561.

[26] M. S. Talamali, A. Saha, J. A. R. Marshall, and A. Reina, When less is more: Robot swarms adapt better to changes with constrained communication, *Science Robotics* [online], vol. 6, no. 56 2021, eabf1416, 2021. DOI: 10.1126/scirobotics.abf1416. eprint: https://www.science.org/doi/pdf/10.1126/scirobotics.abf1416. available from: https://www.science.org/doi/abs/10.1126/scirobotics.abf1416.

[27] C. to Wikimedia projects, *Brownian motion*. available from: https://en.wikipedia.org/wiki/Brownian_motion.

[28] *3pi+ 32u4 robot - standard edition (30:1 mp motors), assembled*, [EB/OL], https://www.pololu.com/product/4975/pictures.

[29] P. O'dowd, *Robotic systems (ematm0053, tb2)*, [EB/OL], https://github.com/paulodowd/EMATM0053_21_22, 2022.

[30] *5.5. line and bump sensors*, https://www.pololu.com/docs/0J83/5.5.

[31] Wikipedia contributors, *Pid controller*. available from: https://en.wikipedia.org/wiki/PID_controller.

[32] P. O'dowd, *3pi_ircommboard*, [EB/OL], https://github.com/paulodowd/3Pi_IRCommBoard, 2022.