COORDINATED MOVEMENT OF MULTIPLE ROBOTS IN AN UNKNOWN AND CLUTTERED ENVIRONMENT

NG WEE KIAT
(*B.Eng. (Hons.), NUS*)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF ENGINEERING

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2004

ACKNOWLEDGEMENTS

This project will never be realized without the guidance of my project supervisor, Associate Professor Gerard Leng. Since my undergraduate final project until my Master research, he has been there to guide me. Throughout the entire span of the project, he has helped me tremendously in finding my focus in the project, and he has been always so willing to listen to my problems and ideas. He has relentlessly met me for formal discussions at least once every week to make sure I have been progressing in the right direction. I am truly amazed by his concern for every one of his students despite his busy schedules and meetings he has to attend. And also not to forget the birthday parties he initiated for the few of us postgraduate students and the lunches and dinners that he treated us!

I will also like to thank my peers Mr Cheng Chee Kong and Mr Low Yee Leong. We have been working together for more than a year and our relationship have grew to more then just colleagues in a common laboratory but also friends who are there to support. The nights that we spent searching and thinking for solutions will be memories well treasured! In addition, they have provided me with a lot of recommendations, insights and helped when formulating the search algorithm and building the robots.

I would also like to express my heartfelt gratitude to Mr Cheng Kok Seng, Amy, Mr Ahmad and Pricilla from the Acoustic and Dynamics Department; the

undergraduate students Han Yong, Vincent, Stephen, Kwang Cherg and Shing Yen for providing me with all the technical and moral supports whenever I needed; Master Graduate Mr Cheo for his advices.

In addition, my gratitude is also for DSO that has sponsored part of the project.

Nevertheless, my family and friends has played important roles in my study. Their encouragements, concerns and support are more than meaningful and heartfelt.

And last but definitely not the least, I thank God, for his grace, and for all his great blessings and love that he never fail to shower on me.

TABLE OF CONTENTS

SUMMARY

The objective of this project is to formulate an algorithm that will coordinate the movement of multiple robots to search for a target inside an unknown and cluttered environment. While doing so they are not provided with prior information of the position of the target and are also not allowed to map the environment. In addition, we will access the time taken for robots to complete their search for different number of robots use and in different sizes of search areas.

The system architecture that we have designed for the multiple robots is decentralized, autonomous, localized and homogeneous. Decentralization allows each robot to do their own processing and decision-makings; autonomy allows the whole robot system to function without human intervention once activated; localization allows the robots to function using only information collected from their local environment; and homogeneity requires all the robots to be built and programmed in the same way. During the search the robots will traverse across the search environment and turn back only when one of the robots reached the periphery of the search environment or when they detected the target.

In order to realize the algorithm, we have formulated four different reactive behaviours for all the robots. The first behaviour is obstacles negotiation, which helps a robot find an obstacle-free path. The second behaviour is homing, which will guide a robot towards the target upon detection. The third behaviour is

flocking, which keeps all the robots close to each other when they are moving. And the last behaviour is migration, which ensures the entire multiple robot system to move in the intended search direction. A robot will only be adopting one of these behaviours at any one time according to the behaviour's importance. Obstacle negotiation is the most important and indispensable behaviour followed by homing, flocking and finally migration. With these four behaviours we developed, the desired coordinated movement will be realized.

We have also studied our algorithm by implementing it on physical robots that we built, and the observations were similar to that observed in simulations. These robots have also been used to performed search experiments to validate the feasibility of the simulation program.

Finally multiple simulations were performed to find out how the time taken changed for different number of robots used and for different sizes of search areas. The number of robots used was varied from three to six and the size of search area from 400m^2 to 1600m^2.

List of Tables

List of Figures

Chapter 1

PROJECT DEFINITION

This project focuses on the problem of coordinating multiple robots to move in a group inside an unknown and cluttered environment in search of a randomly positioned target. In order to achieve this, we have developed an original algorithm to coordinate our multiple robots, and then further tested its effectiveness using our own simulation programs and robots that we built. In addition we have gone on to quantify the performance of our algorithm by analysing the time that the robots took to accomplish a particular target search when the number of robots and size of the environment were changed.

1.1 PROJECT OBJECTIVE

The objective of this project is to formulate an algorithm that can coordinate the movement of multiple robots to search for a target inside a cluttered environment. No prior knowledge of the layout or position of target is given to the robots and they will have to accomplish this without mapping the environment so as to reduce the computational resources involved.

In view of this, there are three main challenges that we have to overcome.

The first challenge is to manoeuvre the robots in the absence of information about the layout of the search environment. This implies that it is not possible to pre-plan the movements of the robots since information of the environment will only be available as the robots explore it. Therefore, real-time decisions will have to be made accordingly.

The second challenge is to coordinate the movement of multiple robots to move in a given search direction in the presence of obstacles. Appropriate behaviours for each robot must be designed so that they will not wander randomly in the search area.

Last but not least, the third challenge is to gauge the overall performance of the algorithm we have designed for coordinating the movement of multiple robots in the search problem. The performance parameter is the time to locate a target and experiments should be performed to analyse how it depends on the number of robots and the size of the search area.

1.2 DEFINITIONS

1.2.1 Search Environment

The environment we are dealing with is strictly two-dimensional and cluttered obstacles are expected. Therefore in order to facilitate possible extensions of our project to future tests involving real life scenarios, we have developed our algorithm with the use of the layout of a typical plantation as the search environment.

According to our survey on existing plantations, the average diameter of a tree's trunk in a typical plantation (as shown in Figure 1-1) is about 0.4m and they are grown approximately 8m apart in regular rows and columns. However, as these dimensions are too big for us to replicate in our project tests, we have to scale it down accordingly. We will scale the plantation according to the diameter of the tree's trunk relative to the diameter of the cylindrical obstacles (0.06m) we are using. This will give us a scaling factor of approximately 6.7 (scaling factor = 0.4/0.06).

Therefore in our project, all the cluttered environments we are going to use for experimentations will have obstacles of 0.06m in diameter and placed approximately 1.0m – 1.2m apart in uniform rows. And hence the dimension of the robots that can be used for the experiment should also not exceed 1m in length.

Figure 1-1

The figure shows a photograph of a typical plantation.

1.2.2 Target

The definition of our target is a beacon that can be identified and distinguished by the robots. In addition, it is assumed that this target will be much slower than the robots.

1.2.3 Search Technique

The method of search that the robots are required to adhere is described in the following steps:

1. The multiple robots are to be released in any randomly chosen direction into the cluttered environment.
2. The multiple robots have to traverse in a fix general direction across the search environment.

3. In order to remain inside the search environment, the robots will have to make a U-turn and move in another randomly chosen direction when they come to the periphery of the search area.
4. All the multiple robots will then move together in this new direction.
5. Go back to Step 2 until the target is detected.

1.2.4 Completion Of Search

In this project the time to search is defined by the time it takes for any 3 robots to locate the target. And a target is located when a robot moves within half a sensor range from it.

1.3 POSSIBLE REAL LIFE APPLICATIONS

In this section we presents two real life scenarios that motivate this research.

Scenario 1: Six Hikers Lost In Mountains of South-western Washington

(Source: http://www.cnn.com/2003/US/West/05/29/missing.hikers/index.html)

A group of three men and three women have gone for a day hike in the Siouxon Peak area of the Gifford Pinchot National Forest, southeast of Mount St. Helens but were reported missing the following morning after some of them failed to show up for work. In the end they were hoisted to safety only on the third day by the search helicopter. For such search mission, it would be extremely useful if upon reception of the report, a group of well coordinated multiple robots were immediately deployed to search the forested area so as to hasten the rescue.

Scenario 2: Terrorist Attack On World Trade Center In United States

(Source: http://www.cnn.com)

After the collapse of the World Trade Center in Wall Street, U. S., damaged gas lines, fires and cascading concrete has prevented rescuers from immediately entering the area to look for the injured and dead. Therefore it would be extremely useful if there were a multiple robot system that was able to search for those survivors in such an unknown and hazardous environment. Even if some of the robots were damaged or destroyed in the process, the rest should still be able to continue if they were made robust enough. And once these robots have found survivors in the ruins, they could send signals back to the base and then further actions could be taken. This would enable the rescue mission to continue without risking unnecessary human lives by sending human rescuers into the collapsed buildings.

1.4 THESIS OUTLINE

The designing of coordinated movement of the multiple robots will be divided into two parts. First we will be formulating the overall architecture of the multiple robot system that will facilitate the search, and second we will come out with the behaviours for the individual robots that will enable them to realize the search technique required.

The contents of all the chapters are summarized below.

Chapter 2 will discuss the related works by other researchers. These works will be presented in two parts – control and architecture of multiple robots, and target search using multiple robots.

Chapter 3 will describe the architecture we have designed for our multiple robots. We will be talking about how we have build a suitable system platform that will enable the coordination of multiple robots to fulfil their tasks of searching the cluttered environment. Our system architecture is one that is decentralized, autonomous, homogeneous and localized.

Chapter 4 will describe the primitive reactive behaviours that we have developed for the robots so that their collective activities can result in their coordinated movement. These basic reactive behaviours are obstacles negotiation, homing, flocking and migration. We will also be providing empirical simulation test in this chapter to gauge the feasibility of these behaviours when implemented on the robots.

Chapter 5 will discuss how we have implemented our algorithm onto a physical platform. A robot that will be used for physical testing will be built.

Chapter 6 will present the tests of reactive behaviours using the robots that were built. Experiments of the physical search will be performed and compared with simulation results to validate the simulation program we developed.

Chapter 7 will show our analysis of the performance of our algorithm when different number of robots and sizes of search area are used.

Chapter 8 will conclude and provide some recommendations for future work.

Chapter 2

LITERATURE SURVEY

The objective of our project is to formulate an algorithm to coordinate the movement of multiple robots so that they can search for a target inside an unknown and cluttered environment. In this chapter we will be giving a review of some of the related works.

2.1 CONTROL AND ARCHITECTURE OF MULTIPLE ROBOT SYSTEM

Due to the potential applications and favourable state of technology, there is an increasing interest in the research on the use of multiple robots. Just as in our project, the means of controlling and organizing them have been the prime motivation behind several of these researches.

One of the earlier works on multiple robots was Reynolds' (1987) object oriented simulation of a flock of birds that were able to move together, split at obstruction and flock together again. He showed that by simulating the behaviours of collision avoidance, velocity matching and flock centring for just the individual birds he was able to create a flock when many of them were replicated.

Beni (1988) introduced a concept of distributed control with his CRS (Cellular Robotic System). This system was made up of heterogeneous units working individually according to their own internal clock, which would collectively produce information for the system based on the patterns formed by their physical position.

These methods of implementing basic behaviour for multiple robot system soon gained more significance as more researches were carried out. Kube and Zhang (1992) used behaviour-based autonomous robots to solve a box-pushing task. The robotic control operated according to five mechanism: (1) using a common task and simple cooperation strategy of non-interference, (2) following other robots, (3) using environment to invoke group behaviours, (4) using other robots to invoke group behaviour and (5) using individual behaviour that was independent of the group. Kube and Bonabbeau (2000) did a more refined study on transportation by ants and replicated the behaviours onto a box-pushing task performed by multiple robots. Arkin (1989, 1992) proposed Schema-Based Navigation to control how robots could be moved to (forage) and retrieved (acquire and deliver) their target by just programming them with the same primitive behaviours, and by doing so they could already organize themselves to collectively achieve their shared goal. He believed that by avoiding a global world model, he could increase the real-time response of the robots. Arkin (1997) also presented the study of approach to multi-agent robotics in the context of two major real world system: (1) ARPA's UGV (Advanced Research Projects Agency, Unmanned Ground Vehicle) that was used for the design of formation behaviours and real-time mission specification with commander intervention, and (2) three Trash-collecting robots that won AAAI Clean-up-office competition in 1994.

Gage (1992) regarded this method of implementing basic behaviours on individual robots as “simple, inexpensive, interchangeable, autonomous”, “rather then through the explicit purposeful, complex, perception-based behaviour of a single very expensive, highly sophisticated unit”. Similarly Parker (2002) programmed her multiple robots with behaviours to solve a multiple moving targets observation problem.

Payton *et al* (2001), Marrow and Ghanea-Hercock (2000), and Kube and Bonabeau (2000), Israel *et al* (1999) being inspired by the self-organizing ability of social insects, have controlled their multiple robots based on the fact that each insect within the colony just has to perform simple and local tasks, and this would result in a more complex global task being accomplished.

The works mentioned thus far have dealt with systems that were decentralized, self-organizing and used local perception to invoke individual behaviour. This means that the robots were required to explore the local environment as they moved and according to the different sets of stimulants from the environment, the robots would decide individually (decentralized control/decision-making) what kind of behaviour it should display. Multiple robots that were coordinated in this manner were also commonly termed as “Swarm” – Beni *et al* (1994) have provided a more rigorous definition. Considering the scenario and requirements of our project, these methods of controlling the multiple robots are indeed very insightful especially when they had achieved their tasks in unknown environment without using extensive computations.

However, there are also researchers that adopted another approach – a centralized controller or planner to command the robots to perform specific actions. Yamashita *et al* (2003) used a centralized control to predict and manage the actions of each of the robots to do complicated forms of transportation tasks. And during the execution of tasks, human interventions were present, but were limited to solving deadlocking situations. Alami and Bothelho (2003) used a planner to allocate tasks to different robots. The strategy was to decompose the mission, decide on allocation of task, allocate the task, coordinate task achievement and finally the actions were executed. Asama *et al* (1991) pointed out that it was easy to implement, made good predictions and ensured collisions avoidance using a central planner because it allowed proper management of all the actions of the robots. However, although centralized approach could make execution of task more accurate and controllable, a critical drawback was that it required significant amount of knowledge from the entire environment before it could work well. Modification might be possible to make this approach more suitable for an unknown environment, but we would expect a concurrent increase in computational resources involved.

Another aspect of discrepancies in the works can be found in the composition of robots in the systems. When there was a common task, the multiple robot system would usually be made up of homogeneous robots (Park and Mullins, 2003; Payton *et al*, 2002; Payton *et al*, 2001; Kube and Zhang, 1992; Arkin, 1990; Reynolds, 1987). For this, all the robots were the same and exhibit the same reaction for the same set of stimulants. On the other hand, if more extensive task allocation was required, a heterogeneous system was sometimes preferred (Matarić *et al*, 2003; Yamashita and Asama, 2003; Gerkey and Matarić, 2002; Beni, 1988). Comparing the methods used in these works with our project requirements, we will probably only require homogeneous robots since all our robots are used for a common search and do not have different tasks or tasks that need further breaking down or reassignments.

2.2 TARGET SEARCH USING MULTIPLE ROBOTS

Depending on factors like the distribution of targets, type of environment, and the quantity and ability of the searchers, different strategies have been used to coordinate robots to carry out a desired search. Gage (1992) and SPAWAR (Space and Naval Warfare Systems Center San Diego, 1998) coordinated the robots into different formations according to three categories of search that they identified: (1) Blanket coverage – to arrange static searchers in such a way that they maximised the rate of target detection within the search area, (2) Barrier coverage – to arrange static searchers in such a way that they minimized the chances of the target slipping through the barrier, and (3) Sweep coverage – to move a group of searchers across the search area in such a way that they could have a balance of the two previous coverage. The search method used in our project was only slightly similar to the category, Sweep coverage, because in our case we required the robots to be mobile rather than static. Gage (1993) has also proposed and made a comparison between a “lawn-mower” search and a randomly wandering search for problems where there were one or more stationary targets placed randomly at an unknown location.

Gelenbe et al (1997) pointed out that the movement of robots would vary for different spatial distribution of mines. For instance, if the distribution of the mines was patchy, then detection of one would mean there might exist more in the vicinity. Therefore the robots should be coordinated to increase its turning rate when the rate of detecting a mine increased. On the other hand, if it was known that the distribution was graded, then the movement of robots should proceed in the direction of positive gradient and at the same time reduce their turning rate.

For more complicated search problems, like searching in a build-up area with elongated polygonal obstacles, La Valle *et al* (1997) suggested encoding the geometry of these obstacles and the spaces into a binary sequence that could be mapped by the robots so that a complete search algorithm could be derived. In this work, the robots, being equipped with omni directional vision sensors, were supposed to ensure that all the targets would lie in at least one of their observable regions. However there were two drawbacks we see. First is the large amount of computational power that we foresee to be required and second is the necessity of precise information of the environment and geometry of obstacles.

Therefore, facing a similar search environment as La Valle, Parker (2002) introduced A-CMOMMT (Alliance Cooperative Multi-Robot Observation of Multiple Moving Targets) as a distributed, real-time method to try to keep targets under observations most of the time. In this approach, robots used weighted local forces to attract them to a target and repelled them from other robots, but it has the disadvantage of robots putting more attention on certain targets.

Wagner *et al* (1999) produced a mathematical model to analyze the performance of Ant-Robots. These robots left traces that decreased in intensity with time, thus when an Ant-Robot moved in an area, which was already divided into tiles, the intensity level of the traces left on each tile would provide the robot with the information it needed to cover the entire area as fast as possible. Park and Mullins (2003) also proposed to use robots to search the area according to their four basic search rules if the search area could be divided into equally distanced nodes to form a grid. These four rules were: (1) moved to the closest node from the current node, (2) if multiple nodes were equally close, the node that was farthest away from the other robots would be selected, (3) if the nodes were equally far from other robots, the one that lied along the same direction just travelled would be selected, and (4) if three conditions were not satisfied, a node would be randomly chosen. These methods of coordinating the robots though might be able to guarantee complete coverage and possibly also achieve the search that we desired, but the question was still how an unknown environment could be divided into the essential grids.

Similarly inspired by ants, Payton *et al* (2001, 2002) used Pherobots that emitted ‘virtual pheromone’ that contained messages. When a Pherobot found a target, it would attempt to broadcast a message to all other robots. And when other robots received that message, they would broadcast again so that the message could be further propagated. As the direction of message received could be known, these robots would be able to form guideposts that would lead to the target. The authors have applied them on two methods of search - the “gas expansion” model where robots would repel from each other, and “bud approach” that was analogous to plant growth where one particular robot would have the strongest repulsive urge. The strategy employed by these Pherobots was a reliable method that could be used to coordinate the movement of multiple robots. However if the distance moved by the robots were large, a corresponding large number of robots would probably be required since more guideposts would be needed. This will not be beneficial to us especially when there is no information of the environment and so the number of robots that may be required may exceed the quantity that we have prepared. In addition, with the search method that we will have to use, the paths of robots will most probably intersect, and so may give rise to additional complications.

In view of our research, the environment of our concern is cluttered and unknown. Therefore it will not be feasible to coordinate robots with well-planned strategies that require the specific layout of the environment and geometry of obstacles. In addition, although the methods of dividing the search areas into node and grids are good ways that can guarantee the realization of a search method, they are not always physically achievable. However, such methods have provided us with some insights to how we can make use of information from the environment to help us improve our search algorithm. For instance, although the entire search environment may not be divided into grids, it may still be possible if we were to limit the divisions to just the local environment.

2.3 CHAPTER SUMMARY

For works related to multiple robot system, popularly used system architectures were the implementation of primitive behaviour for individual robots, decentralized and centralized control/planner, and homogeneous and heterogeneous compositions of robots. As for multiple robots used in target search problems, some of the works utilized the geometry of obstacles or layout of the environment while others divided the environment into grids and nodes for robots' movement. There are also methods that were inspired by self-organizing ability of insects like ants.

CHAPTER 3

DESIGNING SYSTEM ARCHITECTURE FOR MULTIPLE ROBOTS

Over the years, improvements and commercial successes of robots have been noteworthy. Better sensors and microprocessors, lower cost and improved methods of production have allowed robots to be mass-produced efficiently and yet maintain or even improve in terms of their functionality. An example will be the much-publicised Sony toy pet dog, AIBO (http://www.sony.net/Products/aibo/). This robot is capable of exhibiting behaviours that resembled a real puppy, and with the integration of touch sensors and visual sensors etc. it is even able to interact intelligently with its human owners. Some of the other notable examples are Lego Mindstorms (http://mindstorms.lego.com/eng/default.asp), robotic lawnmowers, robotic pool cleaners and robotic vacuum cleaners (Musser, 2003).

However in order to use these robots collectively to perform a shared task, improvements in individual robot's capabilities are not enough. What is needed is a suitable algorithm to coordinate them to work together. Therefore in this chapter we will present a system architecture that we have designed for multiple robots to move with coordination inside an unknown and cluttered environment.

3.1 BIOLOGICAL INSPIRATIONS FOR COORDINATING MULTIPLE ROBOTS

Collective behaviours that are displayed by social animals have inspired us in many ways. Presently we will describe a strategy that is used by ants to self-organize in order for them to achieve more complex goals.

An individual ant that wanders in our house, being physically so small, will probably take a few days just to find a food source lying at the corner of a room. However, in real life, ants do not search for their food individually. If there were really a food source in the room, it will usually not take more than a couple of hours before we notice a line of ants leading to that food source. But how do these small insects managed to achieve this? The answer lies in their ability to properly organize and coordinate among themselves that results in the amazing efficiency in their search for food (Camazine *et al*, 2001; Bonabeau and Théraulaz, 2000; Sudd, 1987).

Camazine *et al* (2001) has done experiments to examine this. The experiment consisted of a colony of ants being exposed to two flat cardboards of different lengths that bridged the nest to a common food source, and these were also the only routes that would allow them to reach their food source. The ants were left to wander freely, and after several hours, it was observed that almost all the ants have started using the shorter cardboard to get to their food source while lesser and lesser ants were observed to cross the other cardboard. The same observations were made when other different length cardboards were used – the ants were always able to choose the shorter path to the food source.

This incredible achievement displayed by the colony is in fact done without central planning or supervision. When the ants leave their nest to forage for food, they will adopt two kinds of basic behaviours. One is that they will secrete a volatile chemical called pheromone, which was deposited along their trails. And second is their instinct to follow the pheromone trails if any. Therefore since pheromones evaporate after a while, the shortest path will be able to retain more pheromone deposits as compared to the longer paths. As a result, most of the ants will eventually converge onto the same path after some time, which will also be the shortest path, to reach for the food and transport it back to their nest.

This leads us to the fact that it may not be essential for our robots' movements to be pre-planned, or for each robot to move in different manner from each other, or acquire global knowledge of the environment in order to realize a search. For the ants, they cannot possibly have information of the entire search area and they have not required central planning to enable them to effectively locate a food source. All the ants just have to deposit pheromones and follow pheromone trails, and a shortest path to food source will be established.

3.2 PROBLEM CHARACTERISTICS OF SEARCH

3.2.1 Unknown Environment

The first characteristic of the search that we have to consider when designing the multiple robot system architecture is the lack of information of the exact layout inside the search environment. Therefore, we are not able to plan the movements of the robots prior to their search. Conversely, the system architecture should facilitate real-time decisions to be made based on information fed back by the robots as they explore the search environment.

3.2.2 Cluttered Obstacles

The second characteristic that we have to consider is the many obstacles that are going to be present inside the search environment. This means that the robots will have to negotiate obstacles frequently, and therefore it will not be easy for them to maintain moving in any fixed order of formation.

3.3 SEARCH TACTIC

As mentioned in Chapter 1, the robots are to maintain in a fixed search direction until any one of them come to the periphery of the environment. Therefore we will make the first robot of the group that reached the periphery generates another random direction that points back to the search environment and broadcasts it to the nearby robots (assuming that our robots can send and receive messages) so that the search direction will change. In actual application, this may be realized by planting some beacons outside the cluttered environment to inform the robots whenever they have moved outside, or if their range sensors can detect sufficiently far, they can also be used to perceive free spaces, which will be used to inform the robots when they were outside the cluttered environment.

Figure 3-1 illustrates how a search scenario will be like when three robots are searching an area. The robots first started off at ‘1’ and moved toward ‘2’. When one of the robots had reached ‘2’ and it generated another random search direction that pointed back toward the search area. This direction was then broadcasted to all the other two robots and they changed direction before they reached the periphery of the search area. The robots were then directed back to ‘1’, and then ‘3’, ‘4’, ‘5’, ‘6’ and ‘7’ accordingly.

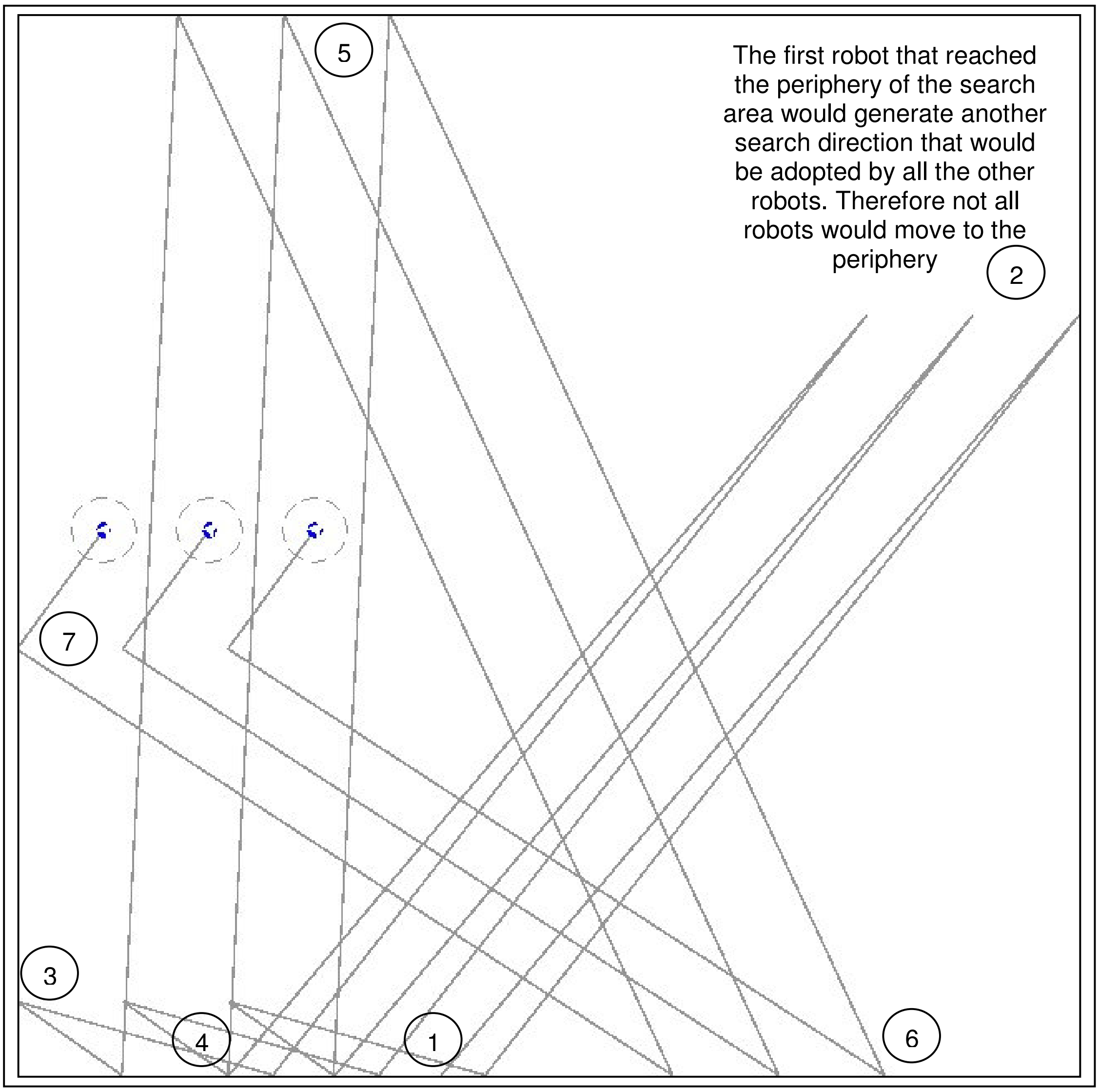


Figure 3-1

A group of robots will traverse the search area by moving together in the same direction until any one of them reaches the periphery of the search area. After that the group will adopt another randomly generated search direction, and the same process will be repeated until the target is found.

3.4 SYSTEM ARCHITECTURE

In order to coordinate the movement of the multiple robots to carry out the search, we have to develop a suitable system architecture for them.

3.4.1 Criteria For Suitable System Architecture

Not only will the multiple robots have to move through the search environment together, they must also allow each member to temporary separate to avoid colliding with the surrounding obstacles. Consequently, to realize these requirements, we will list out the following criteria that a suitable system architecture should fulfil.

1. Robustness – the multiple robot system should allow its members to bifurcate and form back without too much degradation to the proper functioning of the system.
2. Scalable – the system should be scalable in terms of the number of robots in it so that it could accommodate additions, failures or unexpected separation of individual or some of the robots.
3. Simplicity – planning and control of the robotic system should not be too complicated and demands extensive computational resources.
4. Flexibility – the robotic system should be able to manoeuvre through the cluttered environment even if the layout was unknown.

It should be noted that these criteria we have stated are by no means the optimum set of factors to be considered for building any multiple robot systems.

We have decided upon them strictly base on the task and requirements of our project, and they are not representative of general tasks done by multiple robot system.

With reference to these criteria, we will formulate the architecture of our robot system – it is to have a decentralized control, to be able to function autonomously, to perceive the environment locally and to have a fleet of homogeneous multiple robots. The following section will describe in detail these four attributes of our system.

3.4.2 Decentralized Control

The traditional approach to multiple robot control is to incorporate an external central controller that will carry out most of the planning and dissemination of instructions to the individual robots. However, this way of controlling the robots may not be the most appropriate for our project because it lacked robustness and simplicity. Arkin (1992) described centralized control to reduce robustness to the system, limited by communication bottlenecks and excessively complex in terms of designing. Gelenbe *et al* (1997) pointed out that the disadvantages of centralized control were requirement of large amount of computational power and highly complex form of communication. These made such a system rigid and not adaptive. Typically, such a system would need to deal with the complexities in strategic planning as the number of robots changes, and furthermore, there was danger of system overload when the existing processing powers of the robots or central controller are exceeded. Gelenbe *et al* highlighted that it was more appropriate to adopt a decentralized control for multiple robots in which they sensed the environment themselves.

For our task, the number of robots to be used for the multiple robot system will probably have to vary when the size of the search area was different. Moreover even if the number of robots is pre-designated, it will be unreliable to formulate an algorithm for just that number of robots because some of the robots may fail or are destructed during the search. Therefore if a central controller is used, it must be able to deal with the complex task of reassigning the roles and tasks of the robots when their quantity is changed. In addition, this will mean a high dependency on the central controller and if it were to fail, the multiple robots will not be able to continue their tasks until the controller was repaired. Therefore, if a centralize control is used, it has to have high breaking-down tolerance and has to be able to react and re-plan swiftly to adapt to possible change in system size. Moreover, the central controller should also have a very reliable means of communication with the individual robots so that it can send commands and new strategies to them to execute, but again, the unknown environment may not favour steadfast communication.

Hence, we propose to distribute the planning among the units in the multiple robot system by adopting a decentralize control. This is more credible because each robot will be allowed to plan and make their own decisions based on the information they have retrieve. A good example of manifestation of such a system can be found in a colony of ants that is searching and collecting their food

(Camazine *et al*, 2001; Bonabeau and Théraulaz, 2000; Sudd, 1987). There is no centralized control, no leaders and supervising ants to plan, guide and organize the ants in the colony. The ants make decisions themselves, and what emerged is a social group that can effectively defend themselves, multiply and effectively gather food back to their nest. Payton *et al* (2001) have replicated ant's food search ability in their Pheromone robots (or "pherobots"). These robots acted as a distributed set of processors embedded in the environment, performing both sensing and computation tasks simultaneously.

3.4.3 Autonomy

We believe that autonomy can give rise to an increase in the flexibility and adaptability of our multiple robot system because very reliable means of message passing between the system and the human controller may not always be attainable in some environment. In addition, if the robots were built capable enough to make their own decisions, they will be able to react more rapidly to real-time changes as compared to human controllers.

Multiple robotic systems can be implemented with two elementary kinds of autonomy – partial autonomy and full autonomy. A system is said to display partial autonomy when it allows some degree of human intervention during the execution of the tasks. On the other hand, a system has full autonomy when the robots assume total control of their own behaviours.

In most cases partial autonomy is the preferred and more assuring mode of control as it allows necessary human involvement in resolving unexpected deadlocking situations or contingencies that the robots are not programmed to handle. However, for our project, we have chosen to program our multiple robot system to be deployed with full autonomy. This is because we can then try to find out and solve all the foreseeable and likely hindrances to the movement and proper functioning of our search algorithm. In doing so, we will be able to make our multiple robot system more reliable and assuring for future deployment.

3.4.4 Locality

During a robot search, it may not always be possible to obtain global information of the environment. For instance, if our multiple robots were to rely heavily on GPS (Global Positioning System) to provide global information of the environment, it may fail to work properly when deployed to search in the collapsed building mentioned in Chapter 1 because GPS signals may be weak.

On the other hand it will be more reliable for a system architecture that is based on just local information because these information can be retrieved directly by the robots themselves. The ants that we mentioned in the previous section have used solely local perception of the environment to find the shortest path to the food source. Camazine et al (2001) brought out that although unicellular amoebae have very limited sensory range, they were able to self organize to form global patterns based on responds to information in their local environment. Reynolds (1987) created his simulation of flocks of birds by using the aggregate result of each bird that acted solely on its local perception of the world. Therefore, in order to relieve our search algorithm from the dependency on global information as much as possible, we have designed it based on the information obtained from the robots' local perception of the environment.

3.4.5 Homogeneity

The key feature of a homogeneous system is that its units are not differentiable because they are all built and programmed in exactly the same way. There is no hierarchy, no leading robots and no significant difference in composition. Therefore the robots will not need to rely on specific identity of each other in order to execute their own intended behaviours and they will not need to be organized in any specially planned manner in order to carry out their task.

Park and Mullins (2003) mentioned that traditionally it was prevalently perceived that a hierarchy must be present in order to coordinate multiple robots to effectively carry out a task. It was pointed out that a master-slave relationship, although often has enable a particular task to be conducted efficiently, has introduced brittleness into some multiple robot systems. Three major drawbacks with this approach were highlighted. The first involved the communication bottlenecks when a master attempted to coordinate the behaviour of the multiple robots. This limitation reduced the scalability of the system. The second problem dealt with robustness and adaptability. When working in hazardous environment, it was very possible that a single robot, perhaps even the master, might fail, which would then either require the re-election of a new master robot or the redirection of the remaining slave robots based on the number of operational units left. The third difficulty lied in complexity. The designer of such a multiple robot system would be faced with a choice: Design the system asymmetrically (some master and some slaves), with each having different but fixed computational power, or made all robots the same, with each capable of becoming a master agent. The former reduced the robustness of the overall system by limiting the number of robots that were capable of becoming a master while the latter required significant amounts of computational resources to go unutilised when a potential master was operating in a slave robot.

Nature has also provided us insights into the credentials of homogeneous system. Flocks of birds, swarms of locust and schools of fishes etc. are animals that are able to migrate over large distances while foraging for food, taking occasional rests and evading from predators. They have done so effectively while maintaining in groups that have no leaders or hierarchy, in order words, these groups display the attribute of homogeneity.

Matarić *et al* (2003), Yamashita and Asama (2003) Gerkey and Matarić (2002) have chosen to adopt a heterogeneous system architecture, but that was because the task was needed to be segmented and carried out by different groups of robots.

Therefore, since our task does not require further segmentation, we have decided that a homogeneous system will be the most suitable and will bring about robustness in our system and reduces the complication of specific coordination of different individual robots. This will make our robotic system extremely scalability in terms of the number of robots since additional or failures of robots will not require additional effort to reallocate the tasks and re-ordering roles of the present robots.

One additional advantage we can possibly derive from a homogeneous robot system is that since all the robots are the same, they can be manufactured and assembled relatively cheaper via means of mass production operations. This can be an important consideration if we require large number of robots to conduct a search. Furthermore, it is also possible that robots might not be retrievable or are being destroyed in the course of their mission so it will be beneficial if the cost of building a robot can be maintained low.

3.5 CHAPTER SUMMARY

In this chapter we have first talked about the search tactic that the multiple robots would execute. This will require the robots to traverse across the search environment together until the periphery before they make a U-turn into the search environment again. This process will be repeated until the target is detected.

Then we introduced the architecture that we have tailored for our multiple robot system in order to coordinate their movement for a search in the unknown environment. This architecture consists of four features. The first feature is decentralized control, which distributes decision-making to all the members of the robotic system. The second feature is autonomy, which allows the robot system to operate fully by itself with minimal human intervention. The third feature is locality, in which the system will only respond to its local perception of the environment. And last but not least, the fourth feature is homogeneity, which requires all the members belonging to the system to be structurally and behaviourally the same.

CHAPTER 4

DESIGNING INDIVIDUAL ROBOT REACTIVE BEHAVIOURS

In the previous chapter, we have presented the system architecture for our multiple robot system. In this chapter, we will describe how we have made use of this architecture to design the reactive behaviours for the individual robots.

4.1 CONSIDERATIONS FOR INDIVIDUAL ROBOT BEHAVIOURS

4.1.1 Study Of System Architecture

The system architecture that we want to achieve is one that is decentralized, autonomous, localized and homogeneous.

- Decentralized Control – Each robot should determine its own behaviours independently.

- Autonomy – Each robot's behaviours should not require human intervention.
- Locality – The robots would only use local information to aid their decision-making.
- Homogeneity – all the robots should behave similarly given the same set of external stimulants.

One biological system that displays all these characteristics is a school of fish. Balchen (1994) has defined a school of fish as "a collection of a large number of individuals which moves in the water masses as one body with a centre of gravity determining its collective motion and an average diameter as a measure of its geometric extension". This is a biological system that displays self-organizing behaviour without being orchestrated by any central controller (a decentralize and autonomous architecture). The school's behaviour and responses are based on its local perception of the world (localize architecture) and there are no leader or role assignments among the fishes (homogeneous architecture). (Camazine *et al*, 2001) Each fish is simply following four basic behavioural rules: (1) swim away from other fish so that they do not collide with each other; (2) swim near to the school; (3) swim in the same direction as its neighbour; and (4) swim in random direction if it has lost sight of the school or any neighbour. As a result, hundreds of fish can be swimming, migrating, feeding, or escaping from predator and yet at the same time, move together as if they are of one body.

Therefore in our project, the robots will be similarly programmed to just follow basic behaviours that can enable them to collectively move in the intended search directions.

4.1.2 Basic Requirement Of Mobile Robots

SPAWAR (Space and Naval Warfare Systems Center San Diego, 1998) identified five abilities that multiple robots will require in order to accomplish a search problem. These abilities are:

1. To have some form of mobility – this ability is evident since the robot will have to move around in order to find the target.
2. To possess some target-detection sensors – these are sensors that could detect the target. When formulating the search it should also be kept in mind that the range of the sensors is limited.
3. To possess navigational sensor – this ability will determine how robots will measure its position relative to the neighbouring objects.
4. To have some form of communication ability – in SPAWAR, this was mentioned to be optional. Therefore if we were to include this ability during the search, we will be able to use it to pass relevant messages from one robot to another.
5. To have sufficient processing capability – this will determine how complex our search strategy can be.

When we develop the behaviours for individual robots, we will assume that the above-mentioned are also capabilities that all our robots possess.

4.2 REACTIVE BEHAVIOURS

In this context, reactive behaviours are the primitive actions that our robots assume so that the intended search can be realized. We have designed the following four behaviours:

1. Obstacles Negotiation – A behaviour that will ensure a robot will not collide with other objects (obstacles, other robots or target).
2. Homing – A behaviour that directs a robot towards the target upon detecting it.
3. Flocking – A behaviour for the robots to maintain at close proximity with each other. This is to enable robots to carry out certain tasks (e.g. shifting a heavy object that cannot be accomplished by a single robot) together.
4. Migration – A behaviour that helps a robot maintains its intended direction of movement so that it can follow the search direction.

These behaviours above are listed in the order of their importance. And each robot will be choosing one of these behaviours to adopt according to their importance. The selection process is shown in Figure 4-1. As depicted, the most critical behaviour being obstacle negotiation, will override the rest of the behaviours when needed, followed by homing, flocking and migration. For example if the local environmental stimulants require a robot to home a target and negotiate obstacles at the same time, it will have to choose to negotiate that obstacle first.

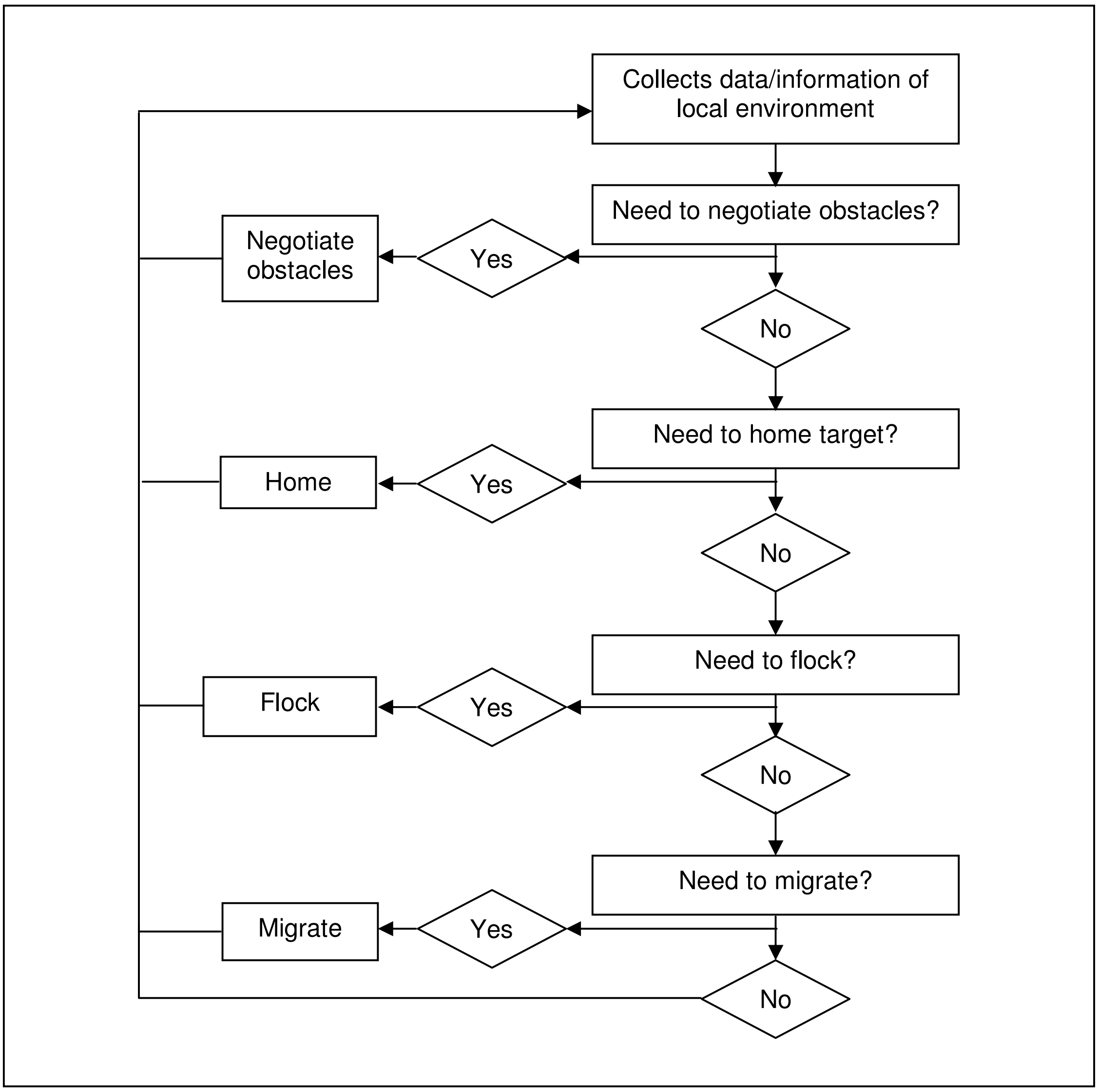


Figure 4-1

The reactive behaviours are obstacles negotiation, homing, flocking and migration. These behaviours are adopted one at a time with obstacles negotiation being the most important to migration being the least important.

4.3 FORMULATING THE REACTIVE BEHAVIOUR

4.3.1 Obstacles Negotiation

Algorithm

Obstacle negotiation is the most fundamental and indispensable behaviour that our robot will need to have in order for it to be able to proceed in an obstacle-free path. This behaviour is triggered whenever the navigational/range sensors detected an object (obstacles, other robots or target) nearer than a specified distance from it.

One popularly used method in obstacle negotiation is the potential field method whereby the movement of a robot is the resultant vector deduced from the summation of the attractive and repulsive force vectors surrounding it. For instance, target can be attractive while obstacles are repulsive, so depending on the magnitude of the forces, the robot will decide its next motion.

However this method will work well only if the actual robot has range sensors that are able to sense continuously around it, but this is not easily realized on physical hardware. In addition, Koren and Borenstein (1991) identified that potential field method often did not allow a mobile robot to move through closely spaced obstacles, and it would tend to cause robots to oscillate in the presence of obstacles or when there was a sudden change in the width of a narrow passage. Borenstein and Koren (1990) have adopted an alternative method called Vector Field Histogram (VFH) that divided the perceived world of the robots into sectors to move the robots. In real-time, the mobile robot would choose the sector that has the lowest obstacle density to move when negotiating obstacles. The authors believed that employing this method would allow a mobile robot to move in an unknown and cluttered environment at high speeds and without oscillations. However their research has been focused on controlling a single robot.

Being inspired by VFH, we assume our robots are equipped with eight equally spaced range sensors, and consequently this implied that a robot will perceive its local environment in eight sectors corresponding to the positions of the sensors as shown in Figure 4-2. However, our obstacle negotiation algorithm will access these sectors according to whether they are blocked or unblocked rather than trying to find the density of the obstacles inside as in the case of VFH. This is because first we can reduce the amount of computation required, and second practical range sensors will usually not enable us to compute the density of obstacles inside their sensing view.

By default, if none of the sectors are blocked, the robot will move in the direction closest to the general direction specified for the multiple robot system (to be elaborated in the later section on *Migration*). However if any of the sectors is blocked, the robot will halt and compute again which sector direction it should move in order to avoid colliding with the object that is blocking the sensor. We will describe the exact obstacle negotiation algorithm using the schematic illustrations in Figure 4-3.

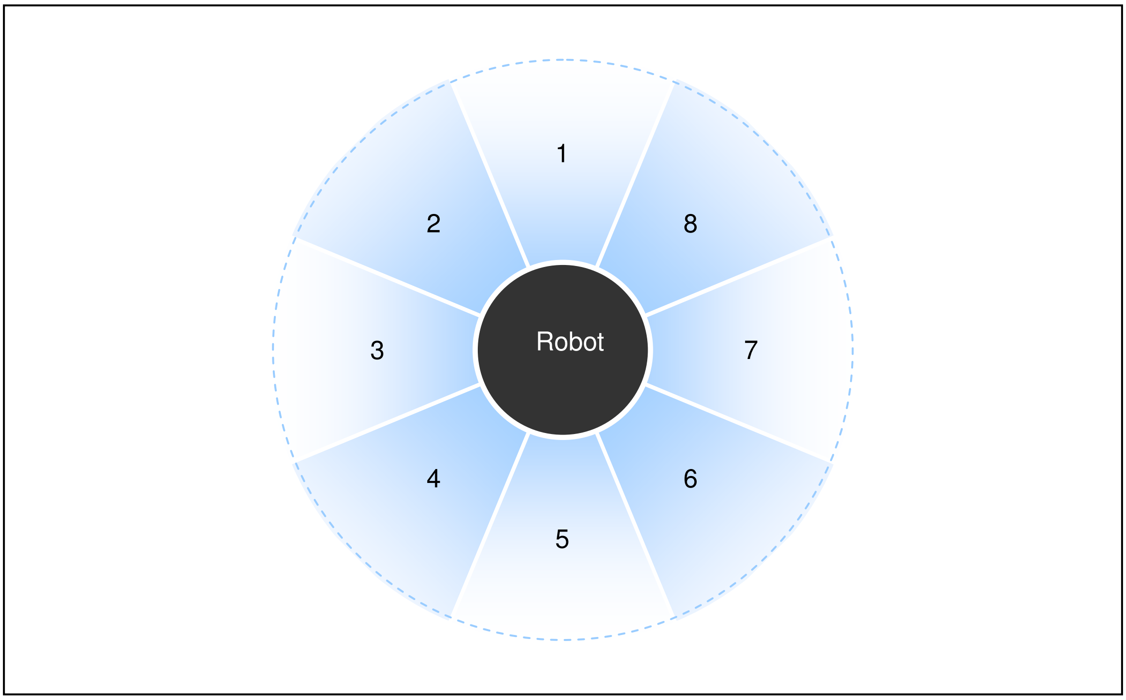


Figure 4-2

The space around the robot is divided into eight equal sectors corresponding to the positions of the sensors.

In Figure 4-3, there are two robots moving in the general direction specified by the white-outlined arrow. When there are no nearby obstacles, the two robots will maintain moving in this direction while the sonar sensors constantly take readings and store them in a one-dimensional array in the robots' memory. However, for the robot on the left of the figure, obstacles have blocked its range sensor 1 and range sensor 8. Therefore, upon receiving this data, this robot will stop moving and begin accessing which range sensors are themselves and their two neighbouring sensors unblocked (these sensors are indicated by the red numberings in the figure). Out of these possible obstacle-free sectors, it will choose the one that makes the minimum angle with the original general direction (white-outlined arrow). The reason to this is to minimize the changes from the

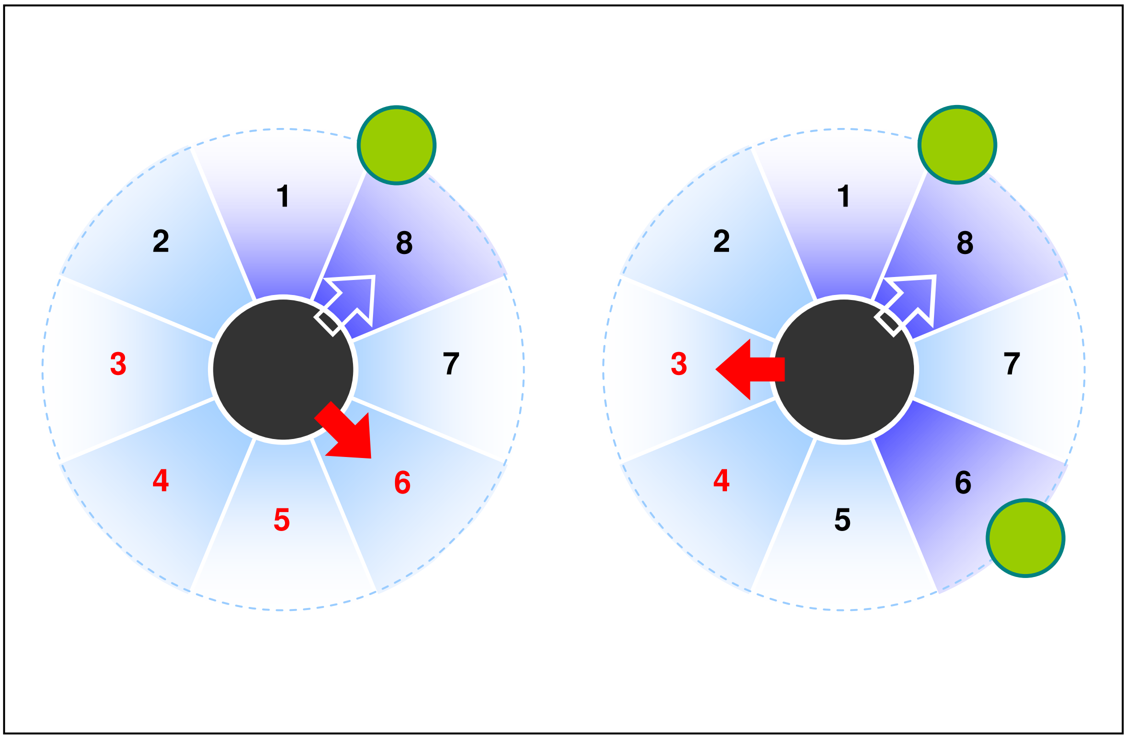


Figure 4-3

The white-outlined arrows represent the original directions that the robot is moving. The red-lettered sensors represent alternative obstacle-free directions. These directions are those that the left and right sensors are also unblocked. The red arrows represent the direction that the robots will eventually choose. These directions are chosen because they make the smallest angle with the original directions (white-outlined arrow).

original direction of each individual robot. Therefore according to this procedure the robot will eventually choose to turn in the direction corresponding to sensor 6 (indicated by the red solid arrow). And after deriving this obstacle-free direction, the robot will move in that direction until all the sectors are all cleared again, or until another different set of sonar sensors are blocked, before it will turn back to the original directions (white-outlines arrow). The sequence is summarized in the flowchart in Figure 4-4.

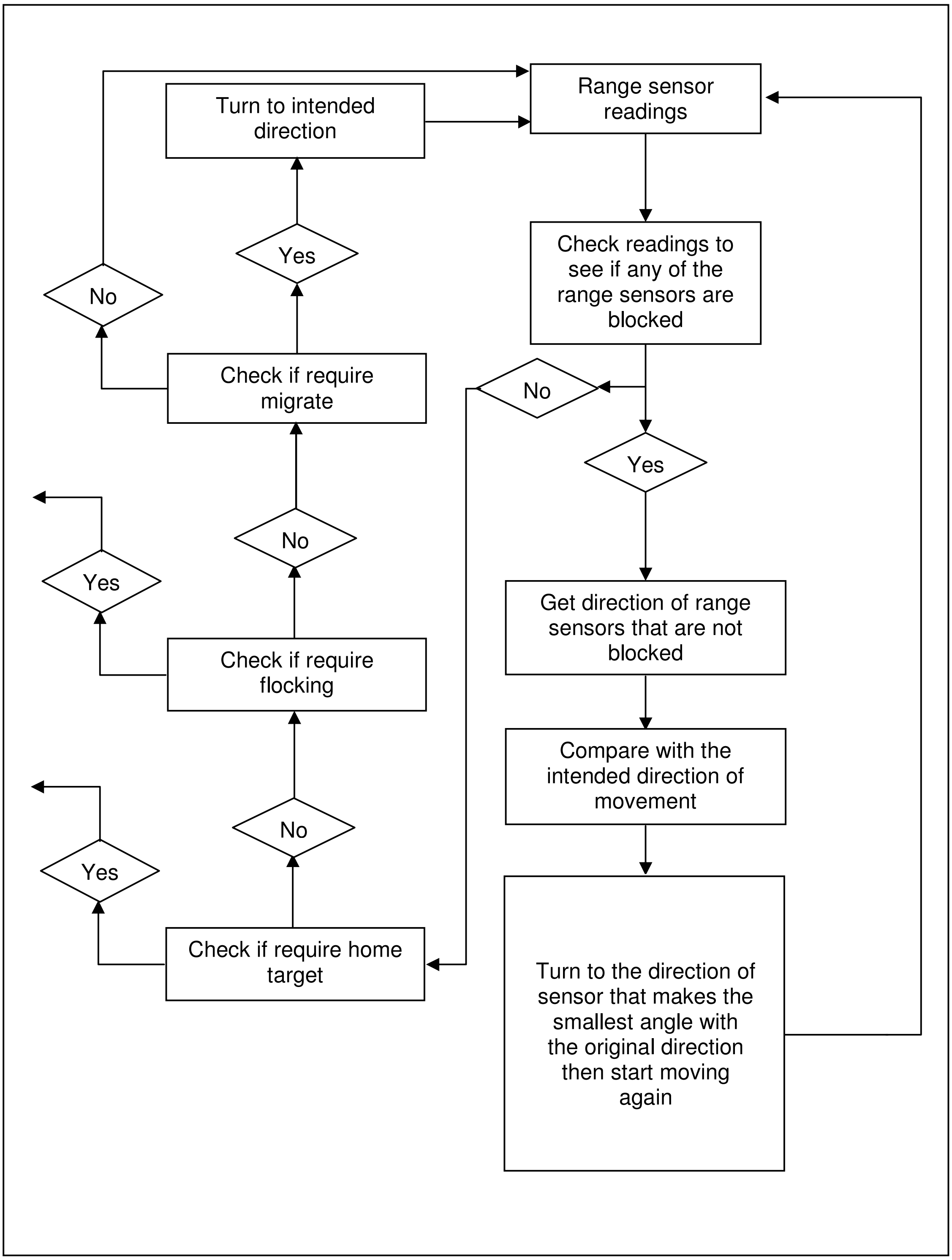


Figure 4-4

Obstacles negotiation and migration algorithm.

Applying the same logic to the other robot on the right of the figure, it will derive at the direction corresponding to sensor 3 to turn.

This obstacles negotiation that we have developed reduces computation requirement and complexity of having to gather large amount of information. In addition, because each robot will be programmed to turn and move only in the eight directions corresponding to the sensors, they will not be required to make minor changes in the directions they are heading too often and hence will still be able to move in the intended direction most of the time.

Implementation On Simulation

Figure 4-5 shows a simulation of a robot (represented by the blue circle with a white arrow) moving inside a cluttered environment. The straight trails left by the robot depict the migratory behaviour of the robot (details will be described in Section 4.3.2) and the curvy trails depict the obstacle negotiation behaviour. The simulation shows the robot repeatedly skirted around obstacles that obstructed its path and then returned to move in the original direction again once it did not detect any obstacle.

Other examples of obstacles negotiation behaviour could be observed in Figure 4-7 and Figure 4-11. In these examples, multiple robots were deployed.

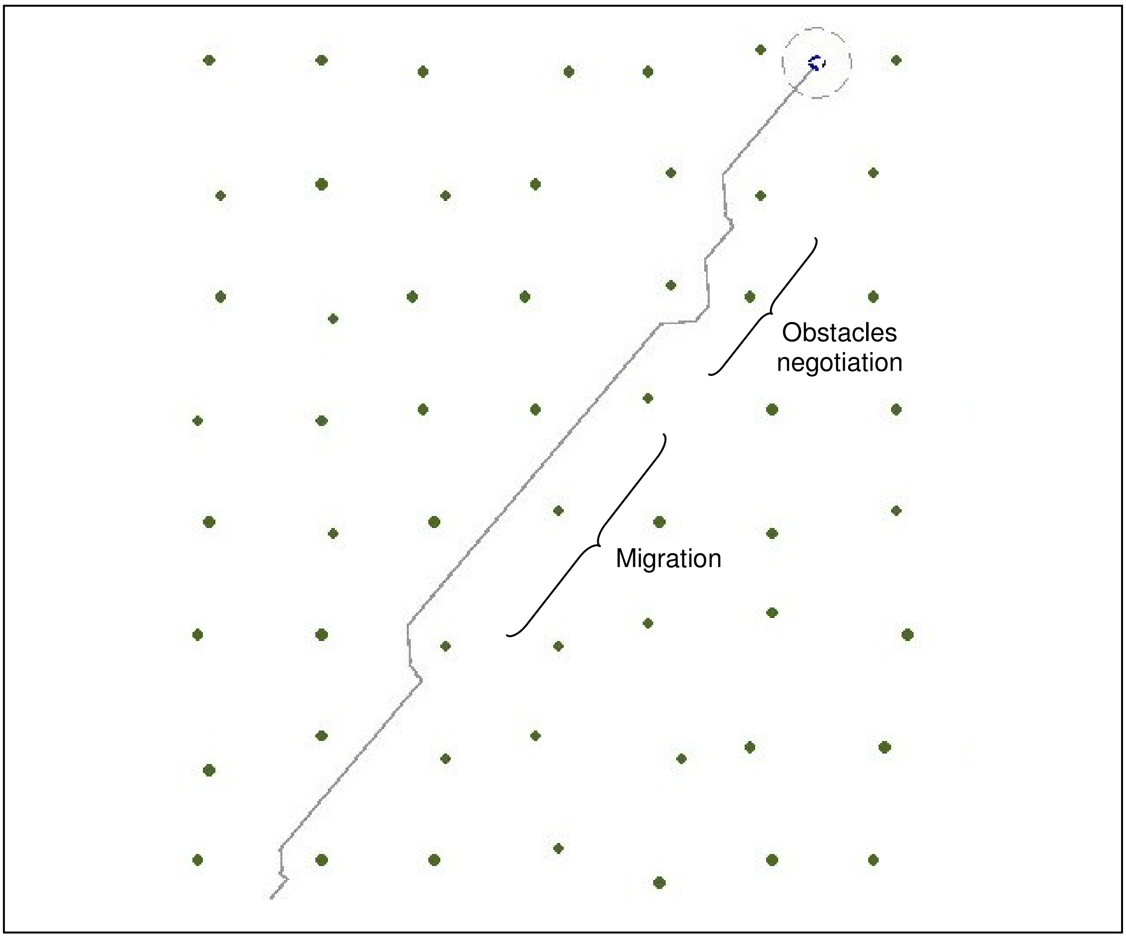


Figure 4-5

The robot would try to maintain moving in the same direction (Migration), veering away (Obstacle negotiation) only when obstacles obstructed it.

4.3.2 Migration

Algorithm

Migration is the reactive behaviour that the multiple robot system traverses in a fixed search direction. This behaviour will not only ensure that the robots move in the desired direction whenever possible, it also provides a preliminary means to minimise the chances of the individual units spreading out too much. This is inspired by migration of birds or fish when they have to travel long distances in the same direction to reach their destinations. During their journey, they may need to temporary change directions to accommodate different situations and environmental constraints, but when ever possible they will still head back in the original direction towards their destinations.

Similarly, when our multiple robot system moves across the search environment, each individual may not be moving in the same direction as the others because they have to negotiate obstacles or perform flocking when the situation arises. Therefore we have formulated and implemented a migration behaviour where each robot will have to keep record of how much it has turned from its intended direction (which is the direction that the entire multiple robot system is supposed to search) so that it can turn back again whenever situations allow. This means that if the robot does not have to negotiate obstacles, does not have to home target and does not have to perform flocking, it will turn back to that intended search direction.

This algorithm is described together with obstacles negotiation flowchart in Figure 4-4.

Implementation On Simulation

This reactive behaviour is integrated in the same simulation as the obstacles negotiation simulation as shown in Figure 4-5. Examining the robot in this figure again, it was initially moving towards the top right corner of the environment. However when it came to an obstacle lying in its path, it was forced to turn away in order to negotiate that obstacle. After it has gotten itself clear from the obstacle, it turned back to that initial direction and started moving in that direction again. Therefore, as shown in the diagram, by adopting the migration behaviour whenever possible, the robot was actually able to displace itself to the top right corner of the search area. Figure 4-11 depicts a scenario of how migration behaviour was interchangeably adopted by multiple robots together with obstacles negotiation and flocking reactive behaviours. It can be observed that the multiple robot system was gradually displacing in the desired search direction.

4.3.3 Homing

Algorithm

When formulating this behaviour, we have again assumed that each robot has eight equally spaced target-detection sensors around it. This is essential because we want our robots to have all-round target detection capability. The complete homing algorithm is described in the flowchart in Figure 4-6.

While the robots are moving, they should be constantly examining the readings taken from the target-detection sensors. The moment any of the sensors detect a target, the robot will broadcast a one-time message to surrounding robots to inform them the direction of the target relative to its own position, before it turns to move towards the target. As for other robots, when they receive such a message, they will only turn and move in the direction that is specified if they have not

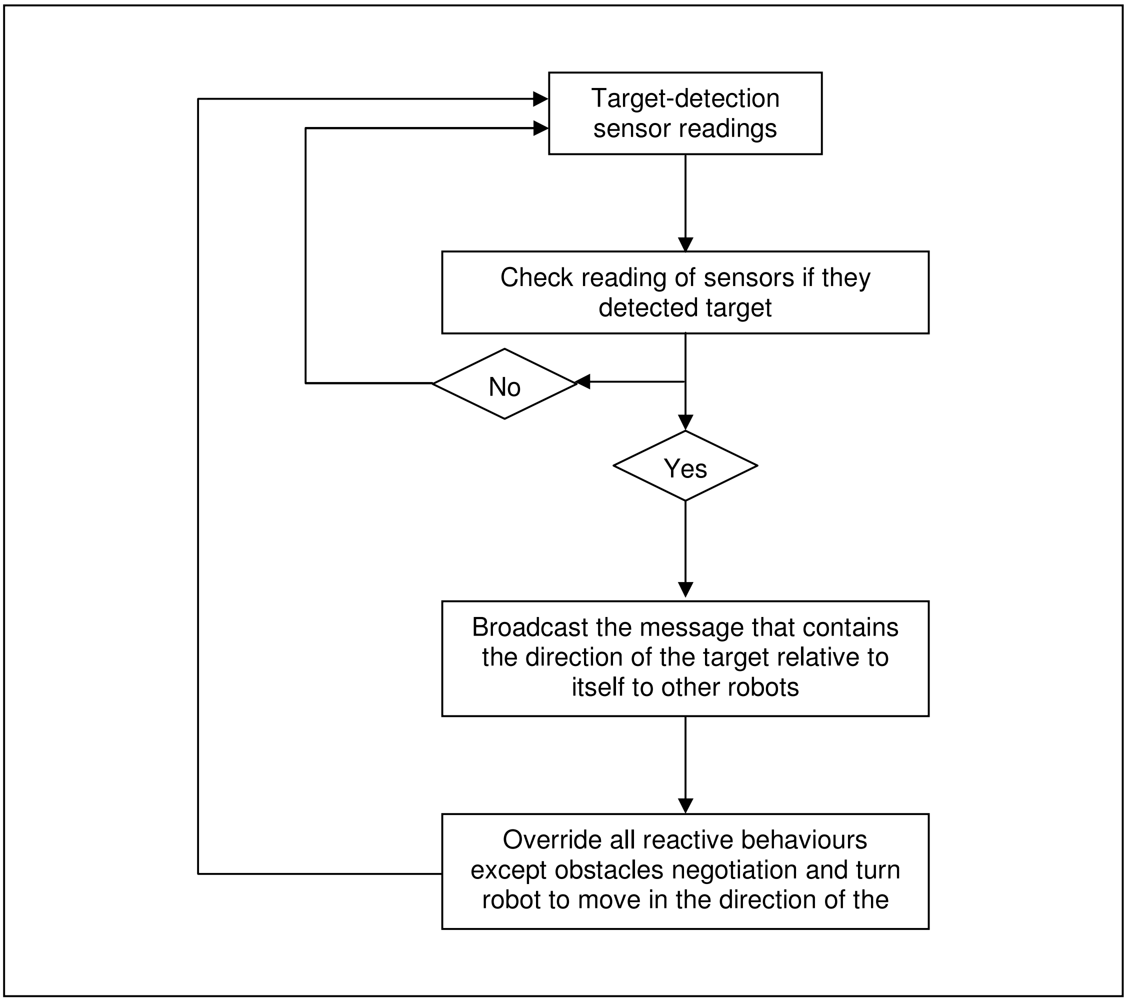


Figure 4-6

Homing algorithm.

detected the target. By doing so, the chances of all the robots detecting and moving towards the target can be increased.

As homing is an essential behaviour that will lead the robots towards the target, it will override all other behaviours (flocking and migration), except obstacle negotiation. This is because we will want the robots to move quickly towards the target upon detecting it, but adopting other behaviours at the same time will tend to slow them down. However, despite this, the flocking requirement will not be compromised since the robots are already close together before that and therefore when they are moving towards the same target location together, they will still be within close proximity with each other even when there is no flocking behaviour.

Implementation On Simulation

Figure 4-7 shows a screenshot of our simulation for multiple robots moving towards a target (represented by the red circle) using our homing behaviour algorithm. For this simulation, the robots are already implemented with obstacles negotiation and migration reactive behaviours. Each robot will perform obstacles negotiation when the obstacles (including target) are less than 0.25m from it, and homing will be carried out when a target is less than 3m from the robot's light sensors. They were initially searching in the direction towards the top right corner of the environment, but the moment one or some of them detected the target, all the robots began to home it, abandoning flocking and migration behaviours. Moreover, it can be observed that although the robots were not adopting any flocking behaviour, they were still within close proximity with each other. This is because during homing, the robots are converging towards the same target.

4.3.4 Flocking

Algorithm

In the previous section, it is said that movement with a commonly shared direction assigned to the multiple robots system (migration) is one way that may reduce

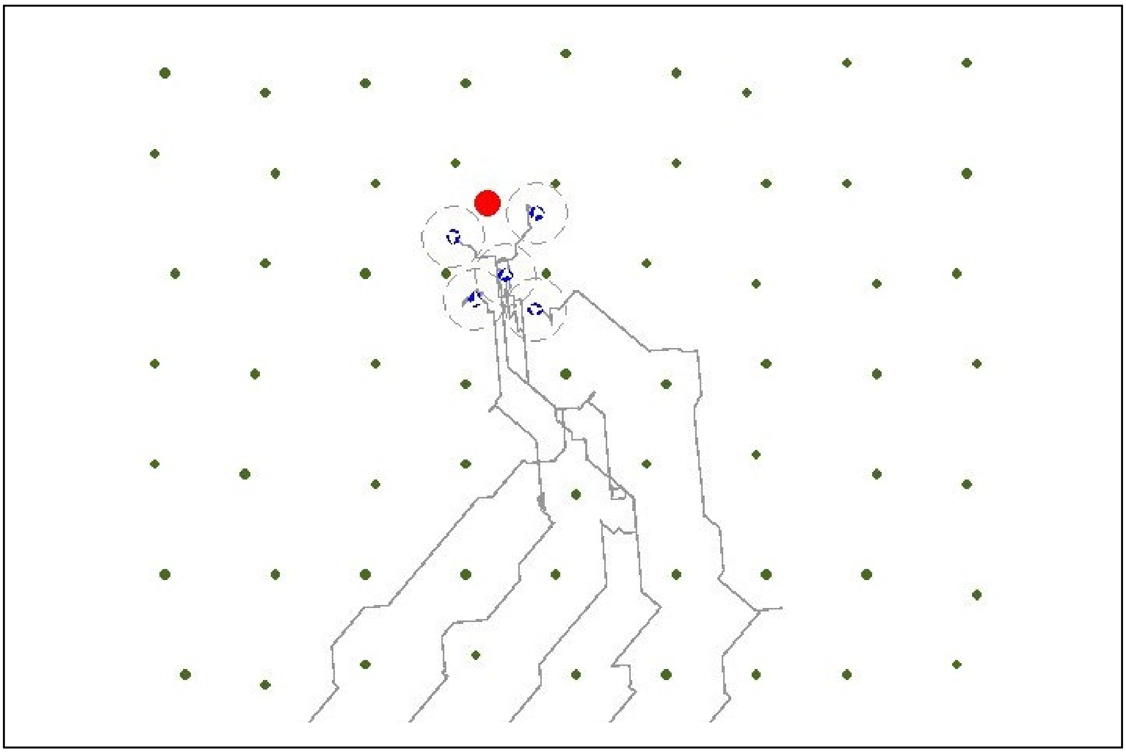


Figure 4-7

The robotic system was supposed to move towards the top right corner of the environment, but it changed its course when its members detected a target (Homing).

chances of robots spreading out. However, having the same direction of motion is still not sufficient when the robots also have to negotiate many obstacles along the way, like in the case of our cluttered environment, because obstacles tend to split the robot system apart. Therefore in order not to let the robots move too far from each other, we have specially designed a flocking reactive behaviour that will enable our multiple robots to bifurcate to negotiate obstacles, and yet purposefully come together again after that. This behaviour will be useful when the size of the search area is very large.

However, before the robots are able to do that, they will first have to be able to identify each other. (Sudd and Franks, 1987; Marrow and Ghanea-Hercock, 2000)

From observation of social insects, the ability to recognize related individuals will set up a more reliable situation for cooperation than will otherwise exist. Therefore we will need to assume an additional type of sensor, robot-detection sensor, other then the two sensors (target-detection sensor and navigational sensor) we have mentioned that will enable the robots to distinguish robots from other objects that are in the same environment.

Assume that the robots now have robot-detection sensors that can be used to identify other robots. We again implement eight of these sensors equally spaced around a robot. Similarly, we will be dividing the spaces around the robot into eight corresponding sectors. However, note that for flocking, we will not design these sectors to be the same as the previous. The composition of the sectors is as shown in Figure 4-8. Now the sectors are not only of unequal size, but there is also a region between the sectors and the robot body. This is a region where robots are sufficiently close enough and they are not required to adopt flocking behaviour. Another thing to note was that these sectors are arranged relative to the direction that the intended search direction (represented by the white arrow). Therefore, if this intended direction were to change, the orientation of the sectors should also change accordingly.

If this robot detects other robots inside its blue sectors, it will move towards that robot in the direction specified by that sector. If there were more than one blue sector being triggered, it will choose the last one according to the sequence that it reads the sensors. This reaction is to make the robot move laterally towards other

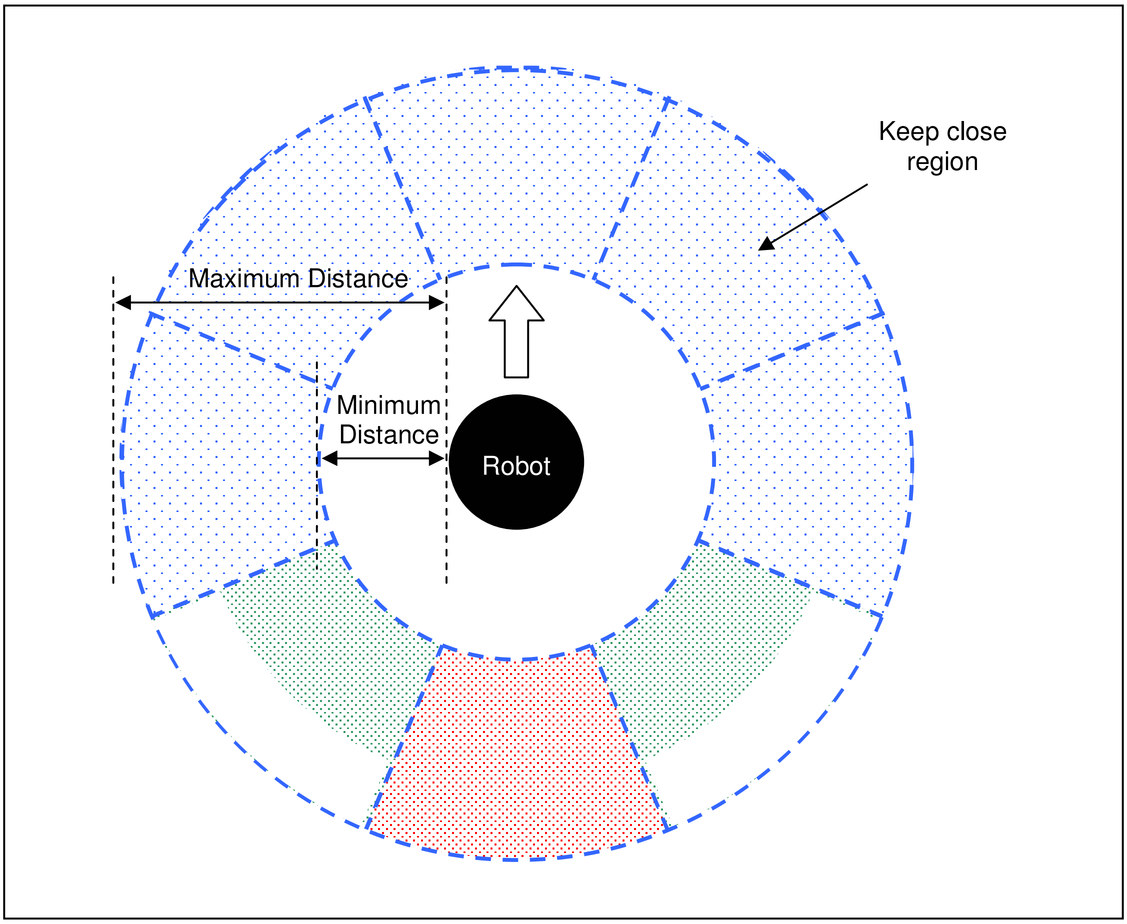


Figure 4-8

A robot will only perform flocking control behaviour when another robot is within the shaded region. In this figure, the white arrow represents the intended search direction the multiple robot system is supposed to follow. Whenever other robots move into the blue region, this robot will turn towards that direction. On the other hand, when there are other robots in the red or green region, this robot will stop moving.

robots at its side or in front of it until they are either out of range (beyond the maximum distance) or close enough (near than the minimum distance).

On the other hand, if other robots come within any of its red sector, the robot will halt at its location. The reason for doing this is for those robots that are too far in behind to catch up. Similarly for the same reason, when the robot detected other robots inside its two green sectors, it will again stop. However, these green sectors have been designed to be slightly smaller than the red sectors. This is because although we want the faster robots to wait for the slower ones, we also do not want them to stop too often and resulting in too much jerkiness to the system's movement. Hence we have designed these green sectors smaller to reduce the frequency of the robots stopping.

Figure 4-9 illustrates an example that explained how multiple robots flocked under assumption of this behaviour. For Robot 1, it would sense robots in two of its blue sectors. Assuming that this robot read the sensors from the one directly in front and counter clockwise, it would eventually choose to move towards Robot 3. For Robot 2, it would sense Robot 1 inside its red sector, and hence it would halt at its current location. For Robot 3, it would move towards Robot 1 since it would sense it inside its blue sector. Finally for Robot 4, as it could only obtain information from its local environment, all other robots would be out of range and hence it would not be required to perform flocking reactive behaviour.

A flowchart of this entire reactive behaviour algorithm can be found in Figure 4-10.

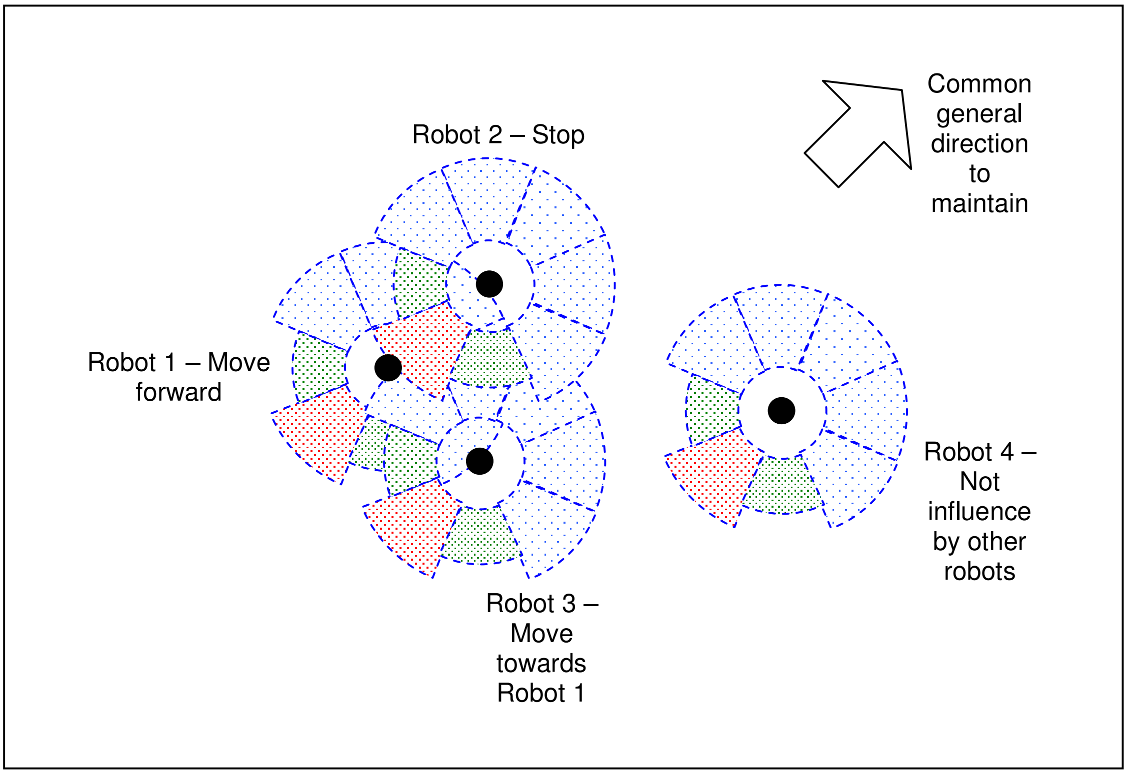


Figure 4-9

When the robots executed the flocking behaviour, Robot 1 would move towards Robot 3, Robot 2 would stop moving, Robot 3 would move towards Robot 1 and Robot 4 was not be influenced by any other robots and would move in the search direction (migration behaviour).

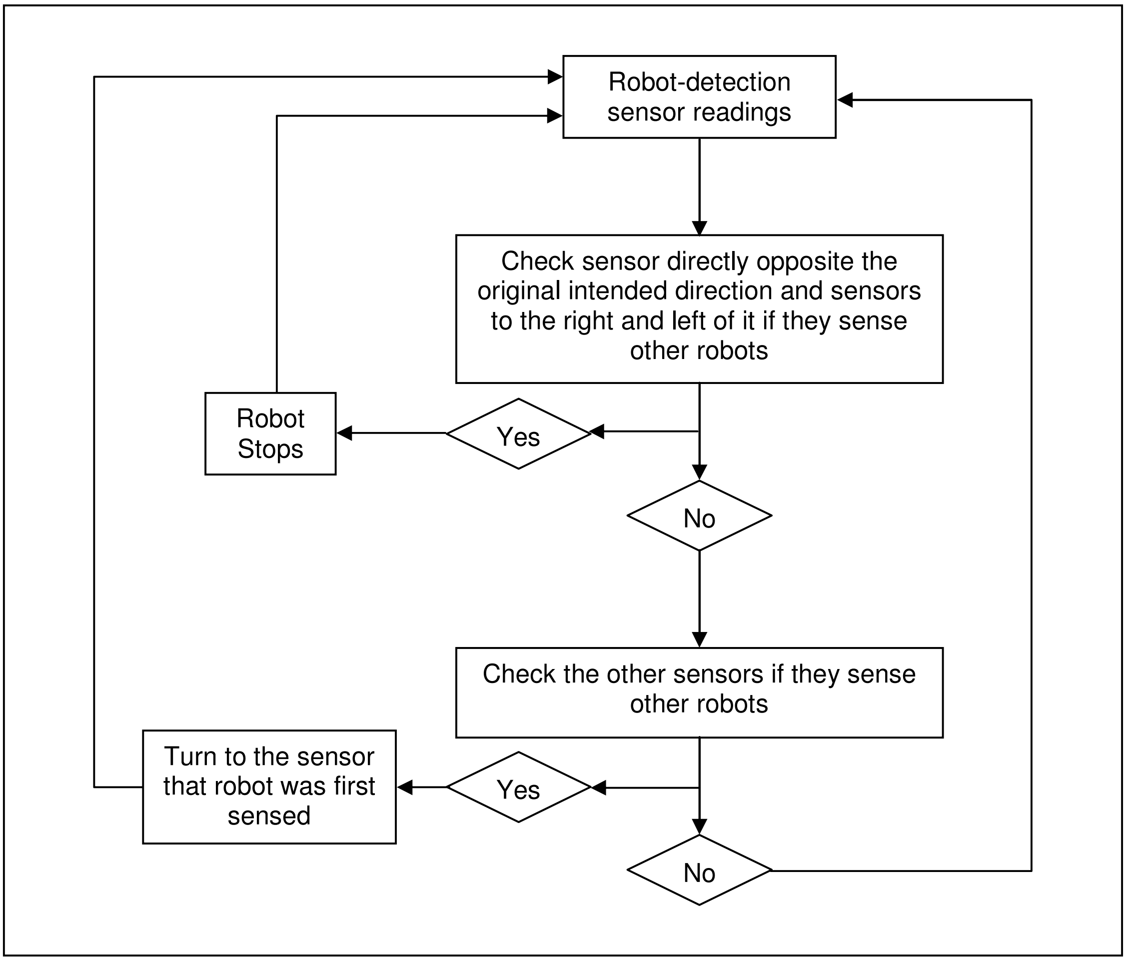


Figure 4-10

Flocking algorithm.

Implementation On Simulation

In this section, we will show the empirical observations of the flocking behaviour when implemented on simulations, and then present multiple simulations experiment that will verify the feasibility of this behaviour.

Firstly, we will present the empirical observation of our flocking behaviour. Figure 4-11 depicts two scenarios. The one on the left showed robots moving without flocking behaviour implemented, and the one on the right showed robots moving with flocking behaviour implemented. It can be seen clearly that when flocking behaviour was present, the robots would tend to purposefully move closer to each other after a while.

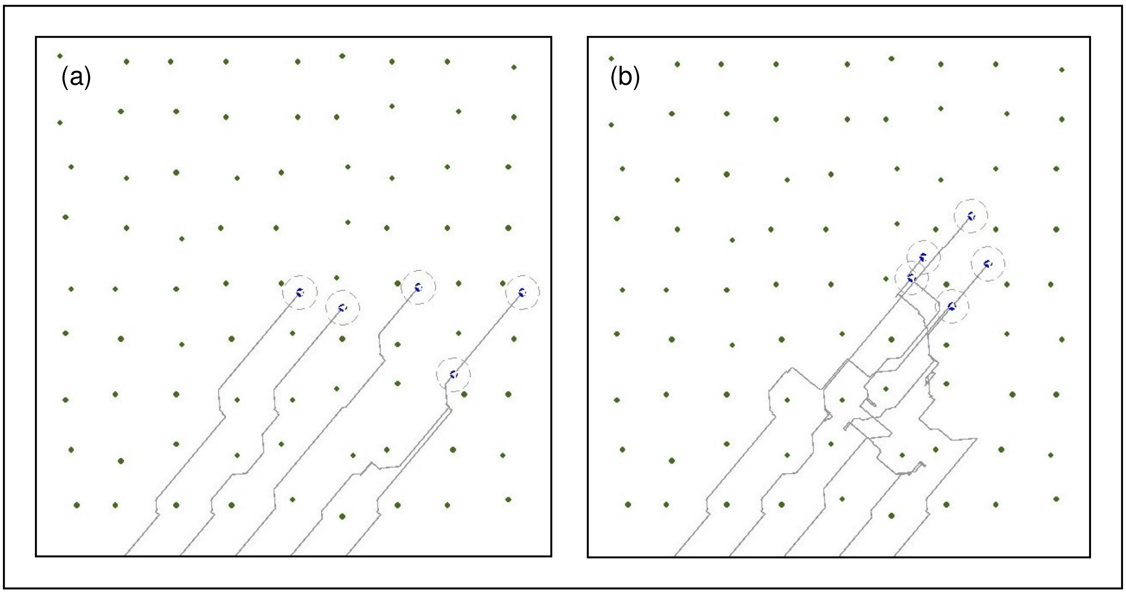


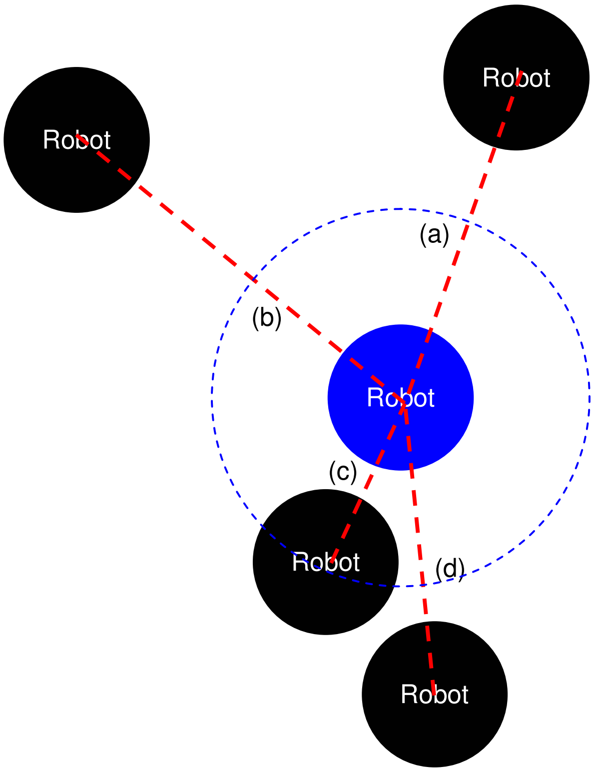
Figure 4-11

Simulation (a) shows a scenario whereby 5 robots were moving in the cluttered environment without flocking behaviour. When compare with simulation in (b), where flocking was implemented, the robots in (a) were relatively farther apart after a distance.

Next we will present our multiple simulations experiments. For these series of simulations, we have recorded and analysed the average distance of the nearest robot from each of the robots in the group. We have done this because the influence of flocking reactive behaviour depends largely on the maximum range of the robot-detection sensors, so unless the robots are able to maintain within that range from their nearest neighbour, they will not be able to perform flocking if we required. In order words, if the robots were to flock properly, then they should always be maintaining distances shorter than the detection range of robot-detection sensors. Figure 4-12 explains the condition for this.

Experiment Setup

In our simulation, we have assumed the maximum range of the robot-detection sensor to be 2m. We have based this assumption on actual sensors that we will be using on our actual robot hardware (details could be found in Chapter 5). Then we carried out fifty simulation runs for each of the multiple robot system with 3, 4,



In this figure, the red dotted lines shows the distances of the black robots from the blue robot, and the blue dotted circle is the range of the robot-detection sensor.

Comparing the distances:
$c < d < a < b$

Therefore if the blue robot is to exhibit flocking behaviour, $c \leq$ (range of robot-detection sensor)

Figure 4-12

Flocking condition.

5 and 6 robots. The robots were allowed to move within a cluttered environment with rows of obstacles placed 1m apart, and their positions were recorded every time they reach the periphery of the search area. For the entire experiment, the average distance moved before the distances of nearest robots were computed was approximately 22m.

Results And Evaluation

The results collected are presented in Table 4-1 and graphically shown in Figure 4-13.

Table 4-1 *Data for flocking*

Number of Robots	Mean Distance of Nearest Robot (m)
3	0.80
4	0.71
5	0.65
6	0.60

These results obtained show that the robots were able to maintain at an average distance smaller than 1m from their nearest neighbours, except in one case when the average distance of three robots was 2.06m. This implies that for the large majority of the cases, the mean distances between the nearest robots have been kept below the maximum range of the robot-detection sensor and hence flocking has been achieved.

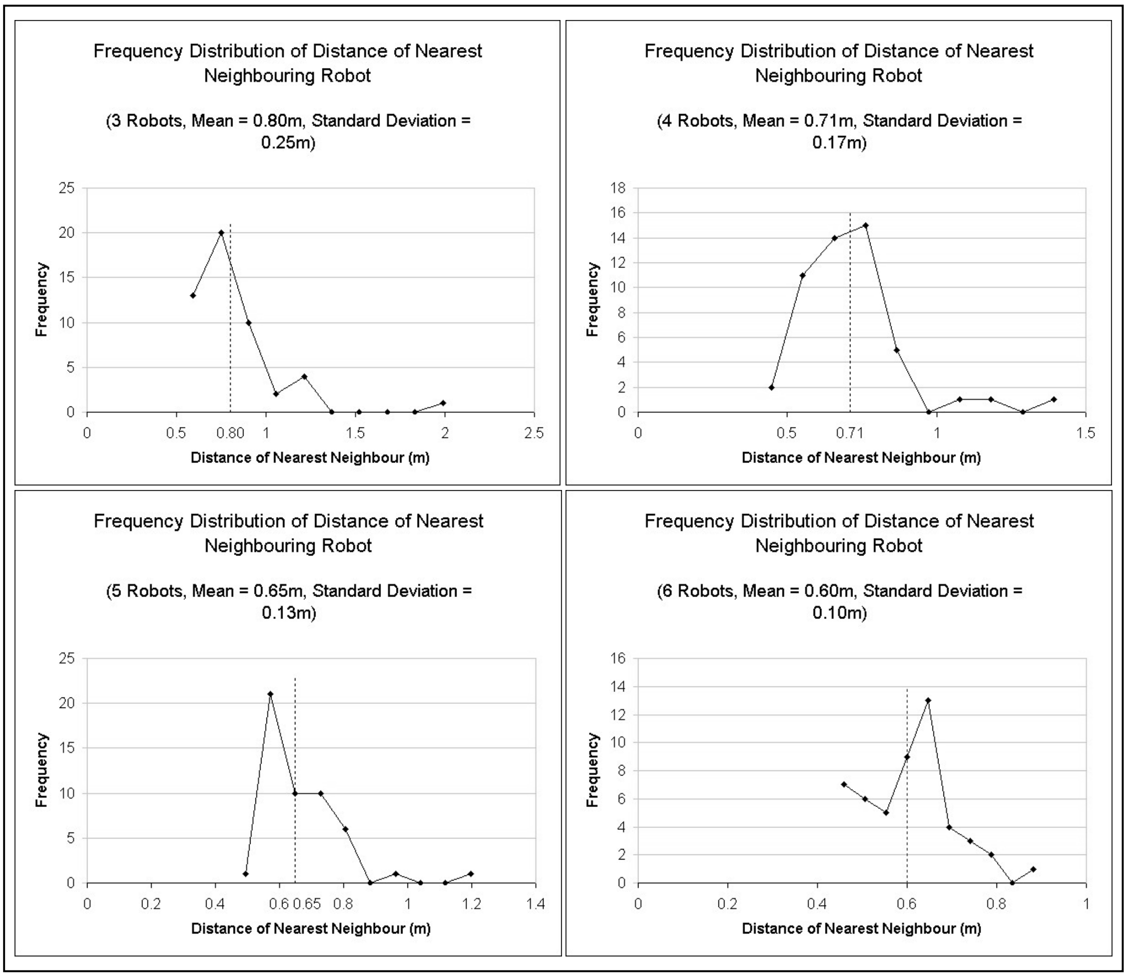


Figure 4-13

The graphs show the frequency distribution of the average distance of the nearest neighbouring robot when a robotic system was traversing in our cluttered environment. The mean distances were 0.80m, 0.71m, 0.65m and 0.60m for multiple robot system of 3, 4, 5 and 6 robots respectively. In order for our robots to flock, they must be less than 2m from their nearest neighbour. Therefore the results show that the flocking behaviour has worked well.

4.4 CHAPTER SUMMARY

In order to coordinate the movement of the multiple robots according to the intended system architecture, every robot will be programmed to follow four primitive reactive behaviours that we have developed. These behaviours are namely obstacles negotiation, homing, flocking and migration, and are listed here according to their level of importance. Obstacles negotiation will enable a robot to move in an obstacle-free path; homing will guide the robots towards the target; flocking will make each robot maintain at close proximity with each other; and migration will help the robots adjust back to the originally intended direction of the multiple robot system whenever possible. At any one time the robot will only be allowed to adopt one of these behaviours. If the conditions are favourable for more then one behaviour to be adopted, it will have to make its choice according to the level of importance of those behaviours.

Chapter 5

HARDWARE IMPLEMENTATION

5.1 CONSTRUCTION OF ROBOT

In the previous chapter, we have assumed that the robots are equipped with the suitable abilities before we formulated the search algorithm. These abilities are namely being able to navigate with navigational sensors, being able to react to the target using target-detection sensors, being able to recognise other robots using robot-detection sensors, and ability of mobility, means of communication and processing capability. In this section we will be presenting the corresponding hardware that we have used to equip our physical robots with these abilities.

1.1.1 Navigational Sensor

When a robot is moving inside the unknown and cluttered environment, it is essential for it to retrieve real-time information about the positions of other objects in order that it can navigate itself. Therefore since the multiple robot system is designed to rely only on local information, each robot will need to migrate in the correct directions and also possess a local coordinate system relative to itself so that it can keep track of the objects around it.

In view of this, we have used two kinds of sensors – one sensor for a robot to know the direction it is going so that it can migrate, and the next sensor for the robot to know the relative positions of the obstacles around it so it can negotiate them when necessary.

Consider the first sensor – we need the robots to know the direction they are heading. This has been accomplished by using Devantech CMPS03 Magnetic Compass. This compass can provide the robots with their current directions relative to the earth's magnetic field.

As for choosing the next sensor, we have to consider the method to specify the obstacles around a robot in the two-dimensional environment. For this there are two possible types of coordinate systems – X-Y coordinate system and polar coordinate system. However we cannot find appropriate hardware that is able to help us achieve a X-Y coordinate system so we have chosen to adopt a Polar coordinate system instead since this can be achieved using range sensors. This is because range sensors are commercially available, and by fixing one such sensor at a definite location on the robot we will be able to obtain the distance and the relative bearing (polar coordinates) of detected objects.

The range sensors we have used are sonar sensors, Devantech SRF08 Ultrasonic Rangers, which are integrated with light sensors. These sensors have range of approximately 0.6m and field of view of approximately 40 degree. Hence it is appropriate for eight of these sensors to be fixed around a robot.

2.1.2 Target-Detection Sensor And Robot-Detection Sensor

The next ability that a robot will need is to be able to recognise objects. In our search environment, there are three types of objects a robot has to differentiate – obstacles, other robots and target. Since our robots do not have behaviours that respond uniquely to the detection of obstacles, no special sensors will be needed in this aspect. However we do require target-detection sensors to recognise the target and robot-detection sensor to recognise the other robots.

Therefore, in our project we have used a fluorescent light bulb to represent the target and have fixed a ring of LEDs, which gave off comparatively lesser intensity light, onto the robots (as shown in Figure 5-2) so that the robots can use the light sensors on the ultrasonic rangers in recognizing and distinguishing the two. This is possible because the light intensity readings obtained from the two sources are distinctly different – the fluorescent light bulb causes the light sensors to return integer values (when we read off via the processors) that are usually above one hundred and the LEDs result in integer values that are usually lower then twenty.

However before we can reliably use this recognition means, we need to dim down the test environment because the normal laboratory light condition actually returns values varying from approximately eighty to approximately one hundred and twenty. Therefore during our testing, we will always try to dim the ambient until the light sensors return a consistent value of zero when there are no other light sources around.

3.1.3 Robot Mobility

The locomotion of our robot is built as shown in Figure 5-1. It is made up of eight ball caskets, a swivelling motor that provides the robot with rotation and a rotating servomotor fixed with a wheel that provides the robot with translation motion.

The intention for such a design is to reduce the change in orientation of the ultrasonic rangers that are housed on the robot body. This is because it has been discovered that these sensors return readings with a delay, which will be magnified if their orientations are constantly changing. Therefore with the ball caskets acting as support and by rotating the swivelling motor, we will be able to minimise the rotation of the robot body when the robot is required to change its direction of motion. In addition, we will program each robot to rotate only to the directions of the ultrasonic sensors so, assuming that each range sensors are the same, the relative position of a robot body to the wheel will remain the same even when directions of motion are changed.

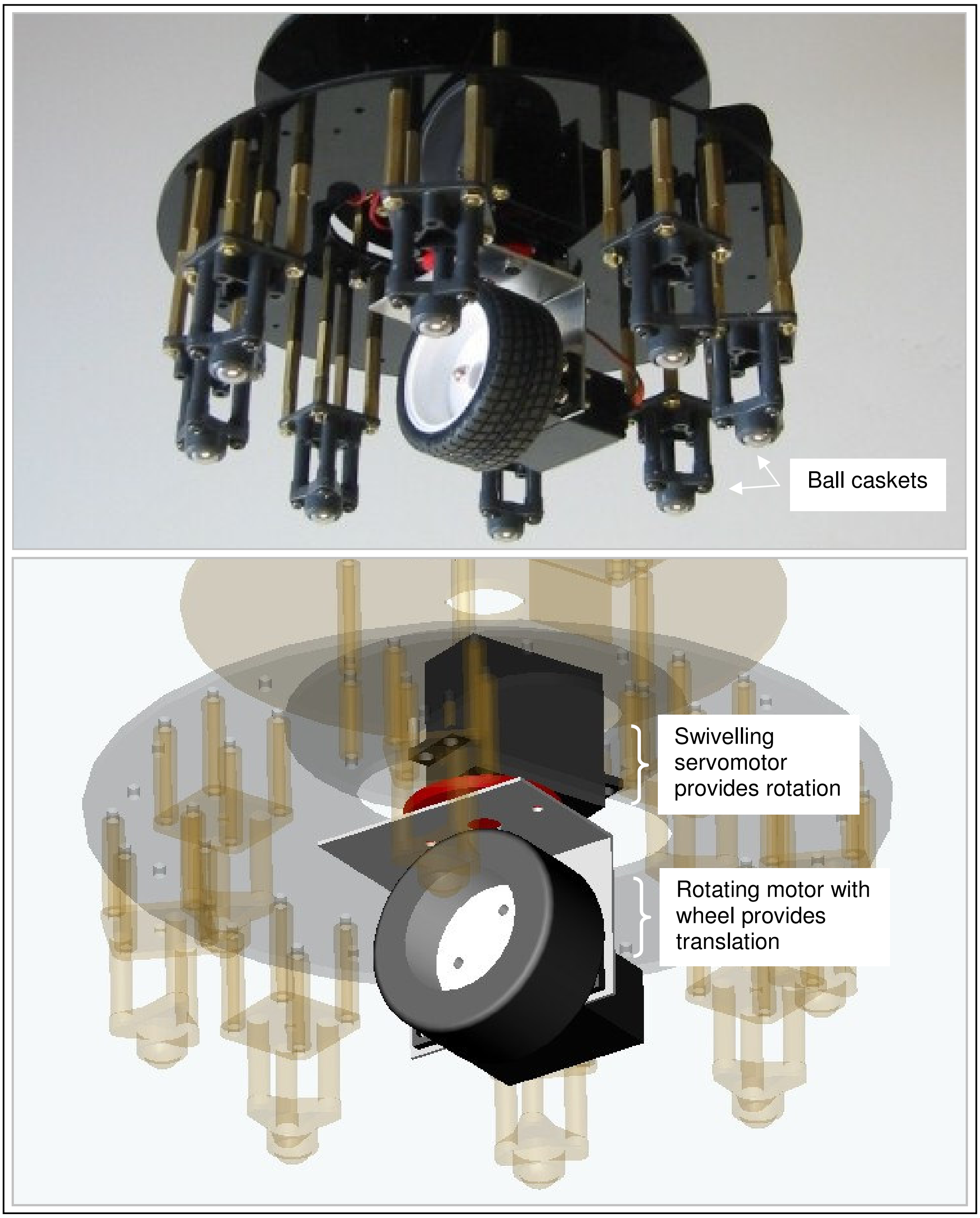


Figure 5-1

Eight ball caskets and two motors make up the locomotion of the robot – one motor provides rotation while the other provides translation.

4.1.4 Processing Capability

All the information retrieved from the surrounding environment is processed in two levels. The first level consists of Acroname Brainstem GP 1.0 that has a 40 MHz RISC processor. This particular microprocessor has been extremely useful because it is readily integrated with a 40 MHz RISC processor, a RS-232 TTL serial port that can tether with a desktop or pocket PC, four servo outputs, a 1MB IIC port that can be connected to the eight ultrasonic rangers and magnetic compass, and digital I/O pins that can be used to switch on the LEDs. However due to the limited processing capacity, Brainstem is used as a slave that is only required to retrieve the data from the sensors and give command to the motors and LEDs.

The second level will be the Master that can request for the sensors data and process them, make decisions and command the robot how to move. This level uses HP iPAQ Pocket PC h5450 (running on Microsoft Windows Mobile 2003, integrated with Wireless LAN 802.11b, 400 MHz Intel Xscale and 128 MB RAM).

5.1.5 Communication Ability

The Pocket PC's provides the means of communication through their Wireless LAN. Therefore we have used this capability to pass messages among the robots.

5.2 FEATURES OF ROBOT

5.2.1 Photograph Of Robot

Figure 5-2 shows a photograph of a robot that we have designed and built.

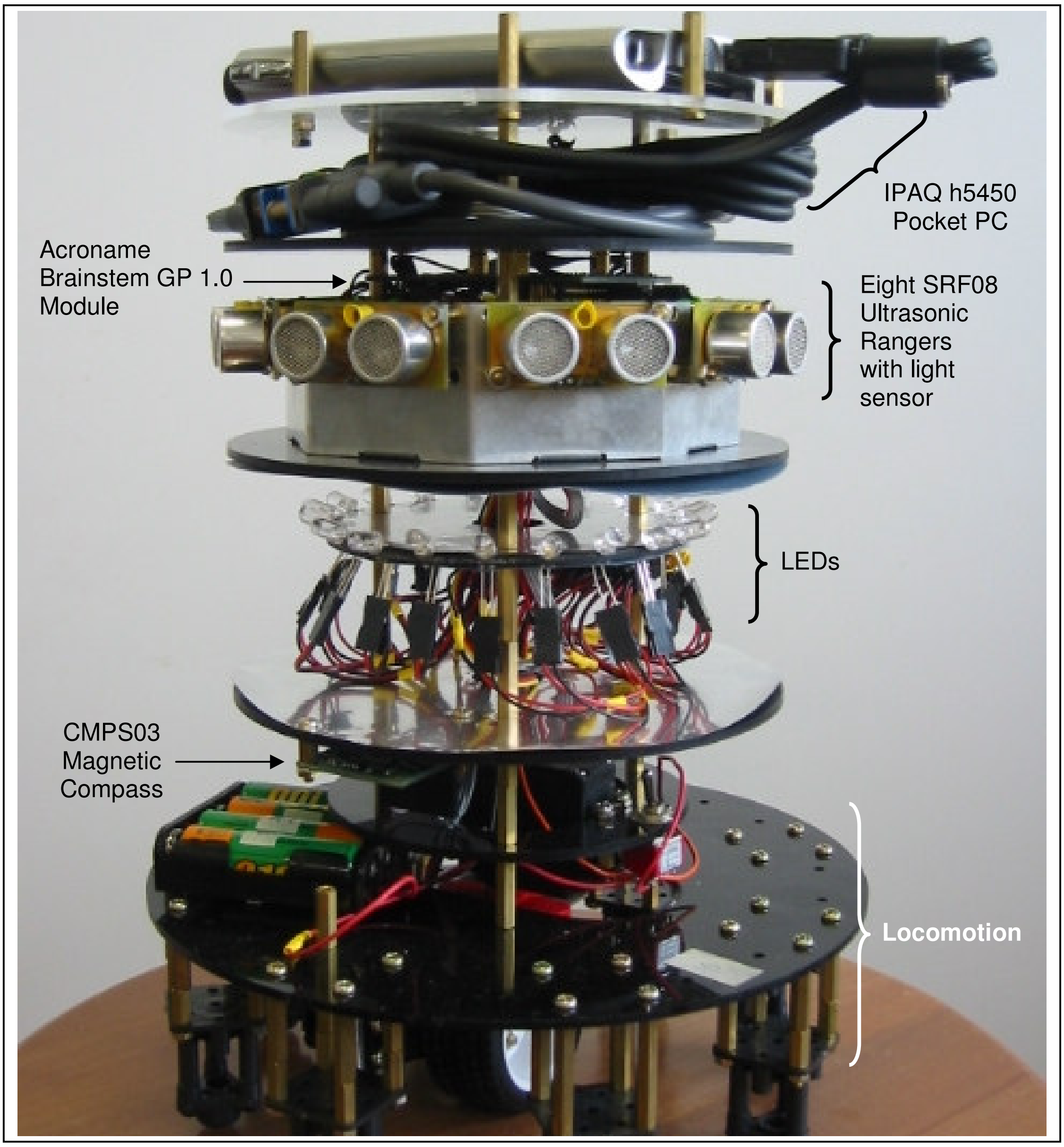


Figure 5-2

A photograph of the robot we built.

5.2.2 Modular Structure

The robot has been designed with a modular structure. In order words, it is made up of structurally independent modules/layers that can provide the robot with different capability. For instance, in our robot there are locomotion module, LEDs module, SRF08 ultrasonic ranger module and processor module. By designing the robot this way, we will have the flexibility of replacing modules or adding modules when needed.

5.3 CHAPTER SUMMARY

In this chapter, we have introduced the hardware that we required to construct a robot for our testing. We have chosen light sensors to be built onto the robots to aid them differentiate obstacles, robots (lit by LEDs) and target (a fluorescent bulb). In addition, we have used ultrasonic ranger to gauge the distance of other objects from a robot. As for mobility of the robot, we have designed a swivelling wheel so as to minimize the change in orientation of the robot body. Finally we have used Pocket PCs together with microprocessors to retrieve data from the sensors, make decisions, give command to the motors and send messages to other robots.

Chapter 6

TESTS ON ROBOT HARDWARE

In Chapter 4 we have shown how individual reactive behaviours of robots are implemented on simulations. Therefore, in this chapter we will program these behaviours onto the robot we built to verify their feasibility in the real world. We will also be carrying out a series of experiments whereby the robots are required to search for target, and compare the time taken with simulated results to validate the simulation program we have created.

6.1 TEST OF INDIVIDUAL ROBOT REACTIVE BEHAVIOURS

6.1.1 Obstacles Negotiation And Migration

For this part of our tests, we will want to see the robots being able to move in the intended direction we specify, negotiate detected obstacles and turn back to the intended direction again after that.

Figure 6-1 shows one of the tests. Note that because we have dimmed down the ambient light during testing, the greenish colour of photograph is produced due to the effect of night vision camera.

The figure shows a robot, which was initially placed to head towards the upper right corner of the environment, moving without being provided with prior information of the environment. Whenever it detected an obstacle, it would negotiate it (obstacles negotiation reactive behaviour) and then when it found a clear path that coincided with the original direction again, it would turn back (migration). From this we can see that the obstacles negotiation and migration behaviour we developed has been able to achieve their purpose.

Comparing this with the simulation behaviour, the movement of the physical robots are slightly jerkier because the flooring condition of our test area has caused the orientation of the robot body to change slightly after it performs the behaviours.

In addition, due to sensor delays and errors, the robots will sometimes move closer to obstacles then expected. However as we have built in tolerance for such inaccuracies, they are still not detrimental to the obstacles negotiation and migration behaviours the robots should display.

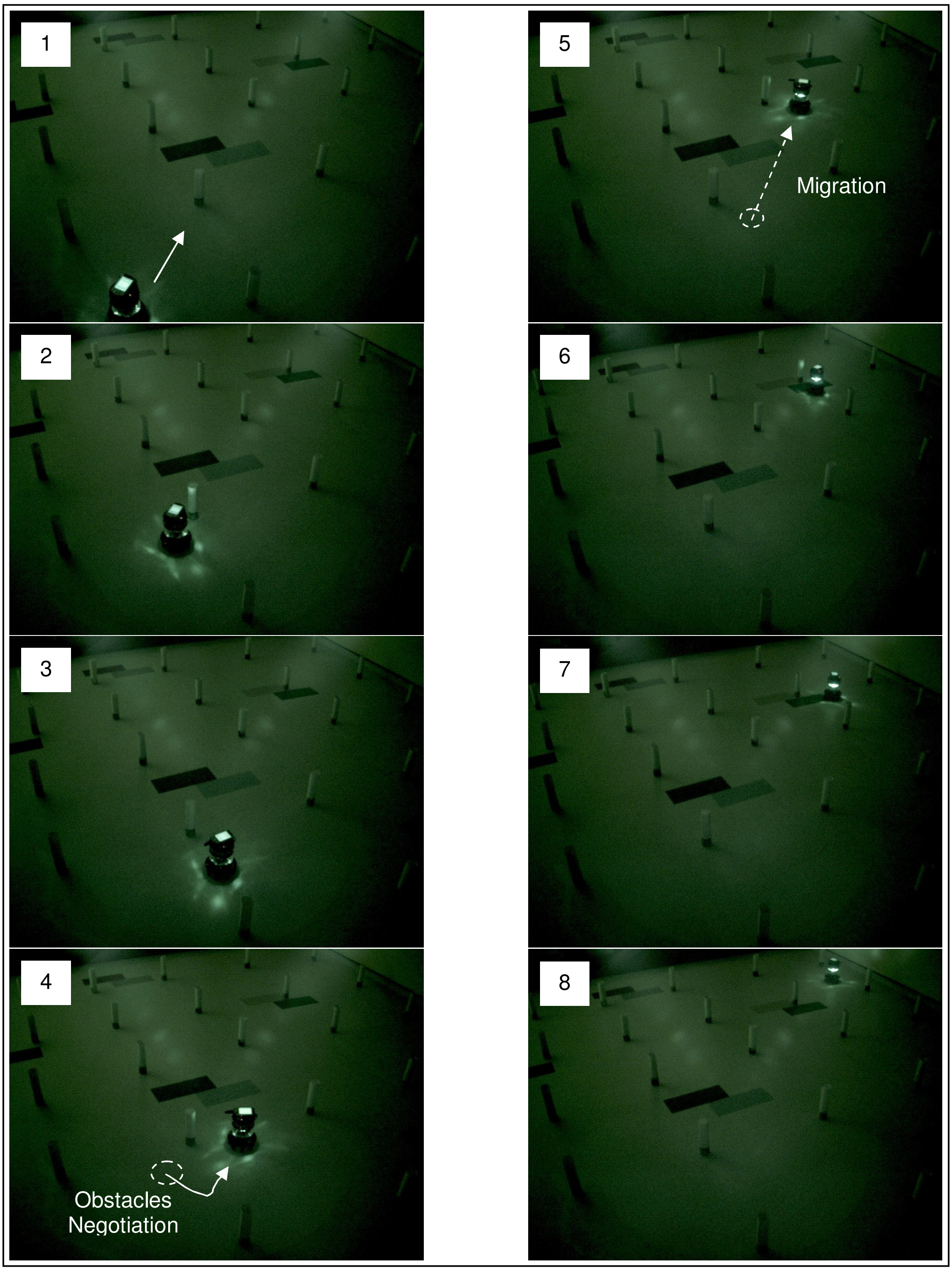


Figure 6-1

Robots displaying obstacles negotiation and migration behaviour.

6.1.2 Homing

For homing reactive behaviour, we will expect the robots to move towards the fluorescent light bulb once it has been detected.

In Figure 6-2, the robots were initially planned to migrate towards top right corner of the cluttered environment. However the moment the robots detected the light bulb, all of them began to change direction and move towards it. This is similar to the simulations and what we will expect our homing behaviour to display.

6.1.3 Flocking

In testing flocking behaviour on our robots, we will be initially placing them apart and make observations on whether they will move closer to each other as they move across the cluttered environment. We will also be carrying out another similar test, but for robots not programmed with flocking behaviour.

The photograph in Figure 6-3 shows what was observed for the two scenarios. It has been observed that when flocking behaviour was implemented, the robots were relatively closer to each other after they have displaced a distance from their original positions. It was seen that flocking behaviour has caused robots that were too far to the left and right of the robot system to move back to the system and for those robots that were too far at the back, the front robot would stop to wait for them to catch up before it began to migrate again.

Comparing again with observations from simulations, the physical robots have behaved similarly, except that the inaccuracies and delays of the light sensors have caused the robots to flock not as immediate as expected, or came slightly closer to each other than expected.

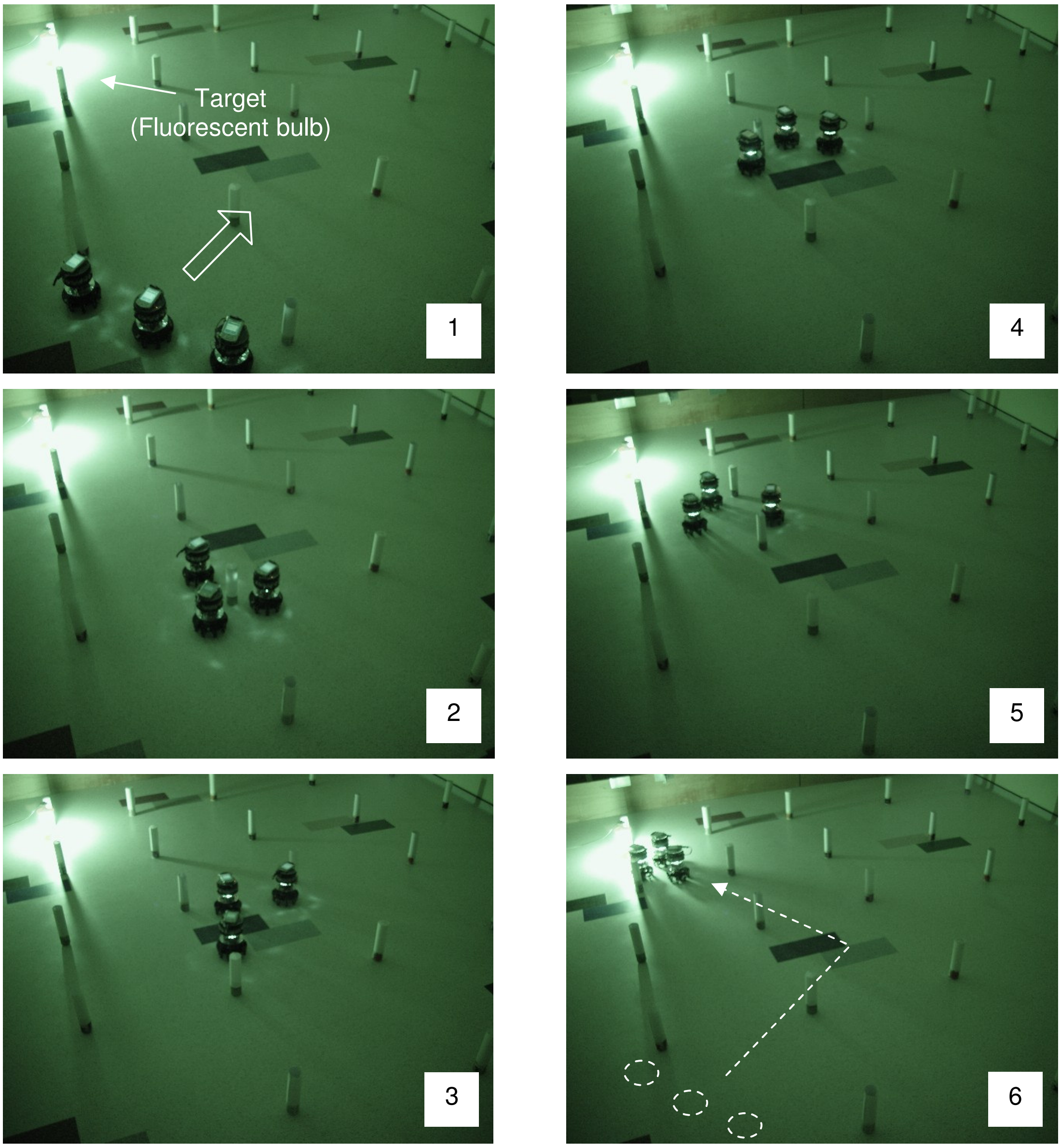


Figure 6-2

Robots displaying homing behaviour.

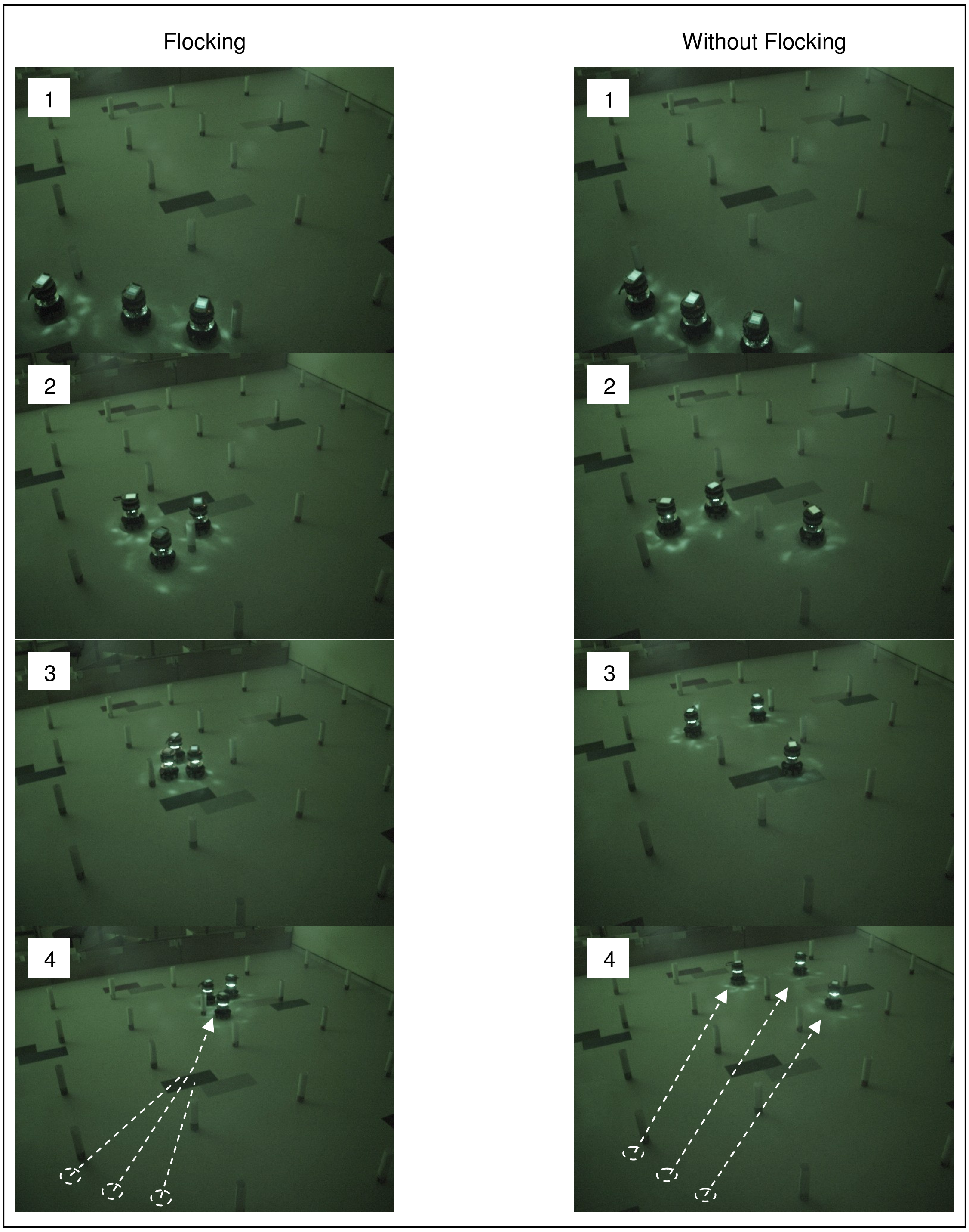


Figure 6-3

Robots moving with and without flocking behaviour.

6.2 TARGET SEARCH EXPERIMENT

In order to validate our simulation program, we have carried out two sets of experiments, namely physical and simulation experiments. For both sets, we have measured the time taken for the search to be completed in environments of the same size using the same number of robots. We have performed a total of two hundred simulation runs to obtain the means and variances of the time taken for a certain search, and we have then compared them with the individual readings obtained from physical experiment for the same search. Our aim here is to validate if our simulation program has depicted the real world close enough.

6.2.2 Experiment Setup

We have set up the simulation environment layout similar to that of the physical experiment environment layout, and the properties of the simulation robot have also been modelled after that of the physical robot. For the search environment, we have used a 10m by 10m cluttered environment that was made up of obstacles that were 0.06m diameter and positioned at approximately 1±0.1m apart in uniform rows. As the search environment was still relatively small, flocking behaviour was temporary removed in these sets of experiments.

Figure 6-4 and Figure 6-5 shows the setup for the physical experiment and simulation. As seen, this physical experiment was done on an indoor badminton court, and the area covered was about 10m by 10m.

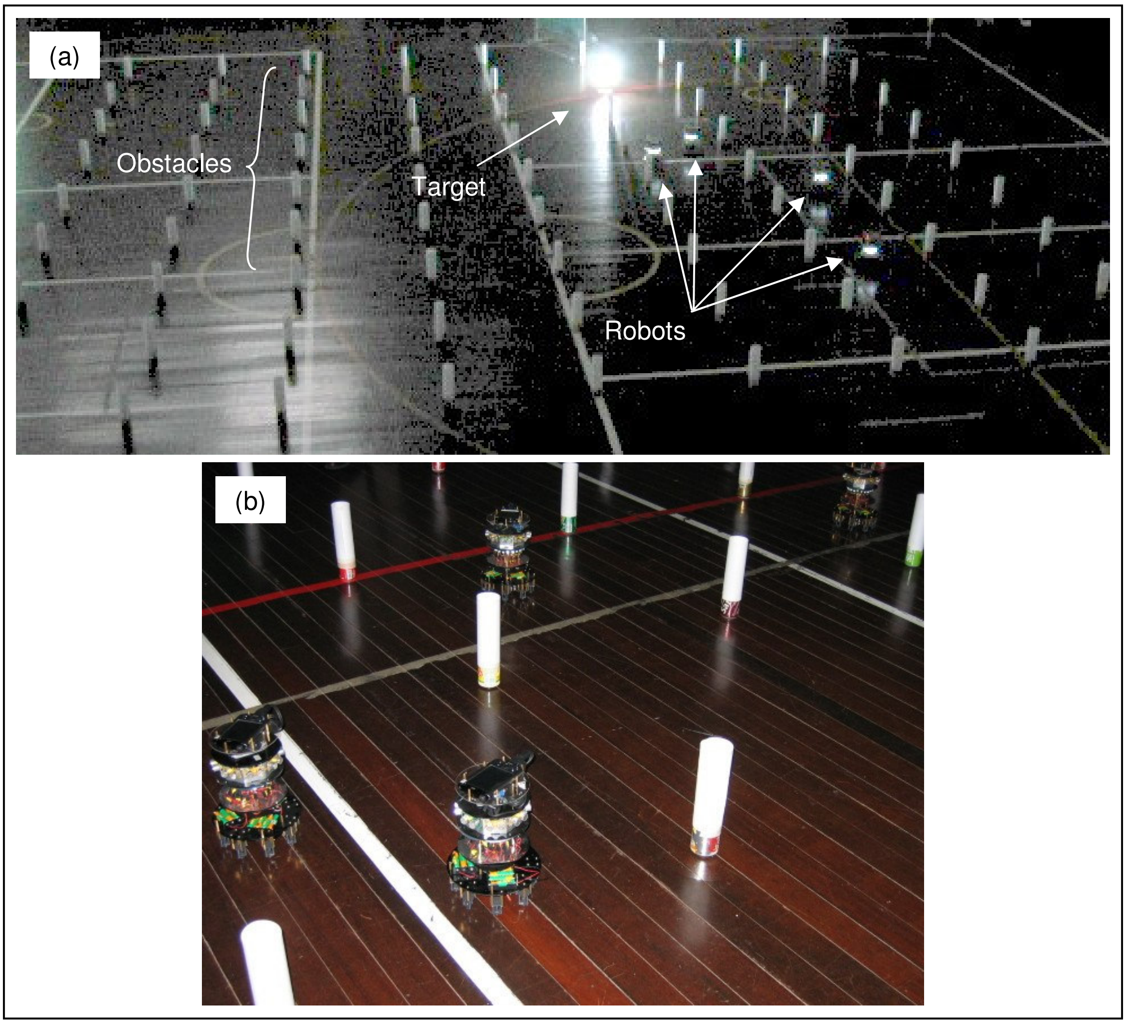


Figure 6-4

Photograph (a) shows our experiment setup with the obstacles placed at approximately 1m apart in rows and columns. The multiple robots traversed the environment in search of the target, and turning back whenever any one of the robots moved approximately 1m from the periphery. Photograph (b) shows a close up view of the robots and obstacles.

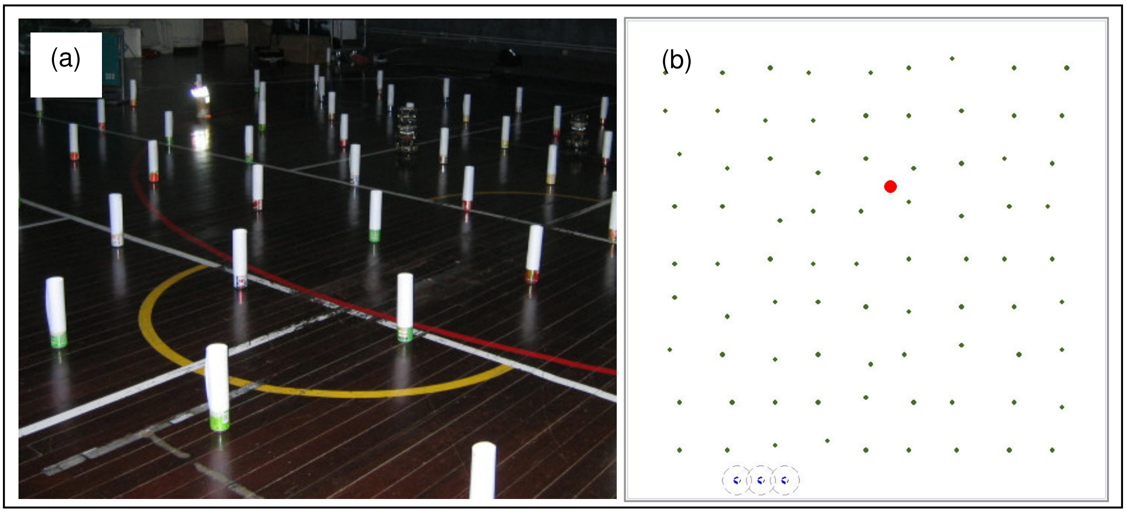


Figure 6-5

(a) Photograph of the physical experiment layout. (b) Screenshot of the simulation layout.

6.2.3 Procedure

For both sets of experiments, the robots were being deployed 0.5m from each other from outside the search environment while the target was placed inside at an unknown position to the robots. A computer had been used to randomly generate the starting positions and directions for our robot system, and the positions of the target for every new run. Then the multiple robots searched the environment according to coordinated movement we designed.

During the experiment, the robots would traverse across the search environment and until any of the robots moved about 1m out from the periphery of the search area, the robot system would have to be directed back into the search environment via commands using Wireless LAN (in the case of physical experiments). Time was taken the moment the robot system started moving until any three of the robots had moved to about 0.5m from the fluorescent lamp.

6.2.4 Results And Evaluations

The results from six physical search and the simulation runs are listed in Table 6-1 and Table 6-2.

Table 6-1 *Data for 3 robots*

	Results from about 200 Simulation Experiments		Physical Search Experiment
Time Taken (s)	Mean = 1439.95	Standard deviation, σ = 2366.90	2280 = 0.35σ
			600 = 0.35σ
			1440 = Mean

Table 6-2 *Data for 4 robots*

	Results from about 200 Simulation Experiments		Physical Search Experiment
Time Taken (s)	Mean = 985.06	Standard deviation, σ = 1464.48	2580 = 1.09σ
			23400 = 0.93σ
			300 = 0.47σ

Due to the nature of such a search, it is expected that the simulation results will produce large standard deviations. This can also be observed from the disparity amongst the readings obtained during our physical experiment. In addition, the flooring condition for the physical environment has been less then ideal. It was slightly undulating and there were potholed regions that have actually caused our robots to temporary get stuck.

Comparing the results, it can be observed that the times taken for the physical experiments we carried out lie within one standard deviation from the mean of the simulations. Therefore we can be 68% confident that the simulations tallied reasonably well with the movement of the physical robot.

6.3 CHAPTER SUMMARY

The reactive behaviours were tested on the physical robots, and they displayed similar behaviour as the simulations we have previously done. In addition, the times taken for the physical robots to locate the target were found to be within one standard deviation from the mean time obtained from multiple simulation results. This validated the feasibility of our simulation program to be used for further experiments.

Chapter 7

PERFORMANCE ANALYSIS OF COORDINATED MOVEMENT

In this chapter, multiple simulations will be done for the coordinated movement of the robots searching inside the cluttered environment. The aim here is to analyse the overall performance of the algorithm we have designed. To do so, we will quantify its performance by measuring the time taken to complete the search for different number of robots and different sizes of search area.

7.1 SIMULATION PROGRAM

We have created a Window-based GUI (Graphics User Interface) simulation written in C++. It was created to allow us to perform more intensive experiment runs and graphically depict how the robots would behave in the search environment. It is the same simulation program that was used previously.

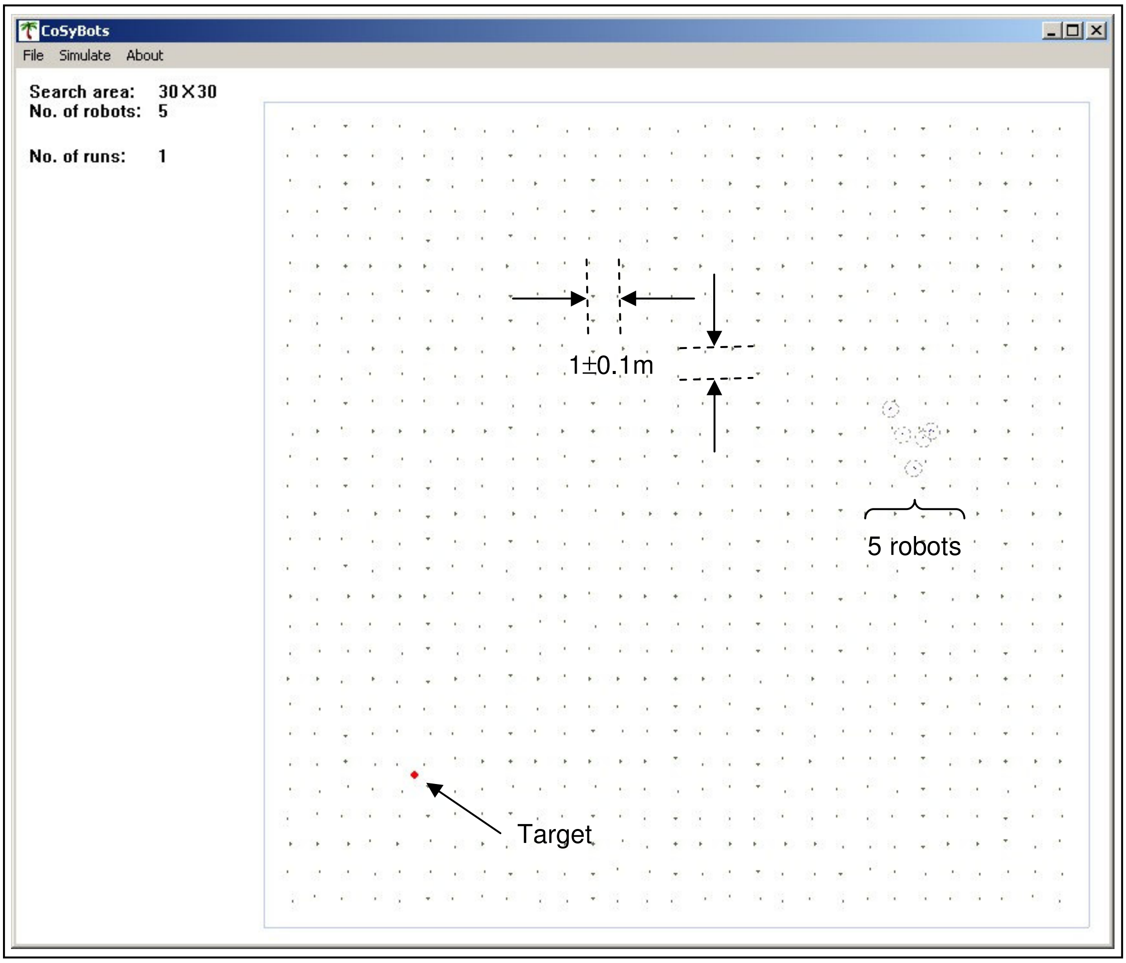
By using object-orientated programming, this simulation program consists of classes of sensors and of our robots that will enable us to realize our algorithm. In addition, in order to better simulate a real world situation, these classes have been modelled after the actual hardware that we have used to build the physical robot.

With this simulation program, we have been able to carry out multiple experiment runs in bigger size search area and using more robots then what we are able to physically build.

Figure 7-1 shows a screenshot of a simulation run.

7.2 SIMULATION SETUP

Similarly, as seen in Figure 7-1, the cluttered environment was made up of obstacles 0.06m in diameter and placed at approximately 1±0.1m apart in uniform rows. The multiple robots were started with their initial positions outside the search environment while the target was randomly placed inside.



Search area: 30×30
No. of robots: 5

No. of runs: 1

Figure 7-1

Simulation run for 5 robots in a 900m^2 cluttered search environment.

7.3 EXPERIMENTS FOR DIFFERENT NUMBER OF ROBOTS

7.3.1 Procedure

For this set of simulations, the number of robots was varied from three to six for the same search area. Our intention is to find out how the time taken to complete the intended search will vary when different number of robots is deployed.

The robots were initially placed 0.5m apart at the periphery of the search area and heading in the same direction into the search area. These initial positions and directions, and the position of the target would change for every new simulation run. Nearly two thousand simulation runs were done to obtain the mean times taken to complete the searches for different number of robots.

7.3.2 Results And Evaluations

The results have been plotted on the graph shown in Figure 7-2. It illustrates how the time changed when the number of robots was varied for the given search area.

For the search areas of 400m^2 and 900m^2, there were relatively little changes in the times taken when the number of robots changed from three to six. On the other hand, when the search area was further increased to 1600m^2, a notable decrease in time taken (approximately 1260s faster in completing a search for an additional robot) was observed when the number of robots was increased from three to five, and a suddenly increased in time taken (approximately 2900s slower

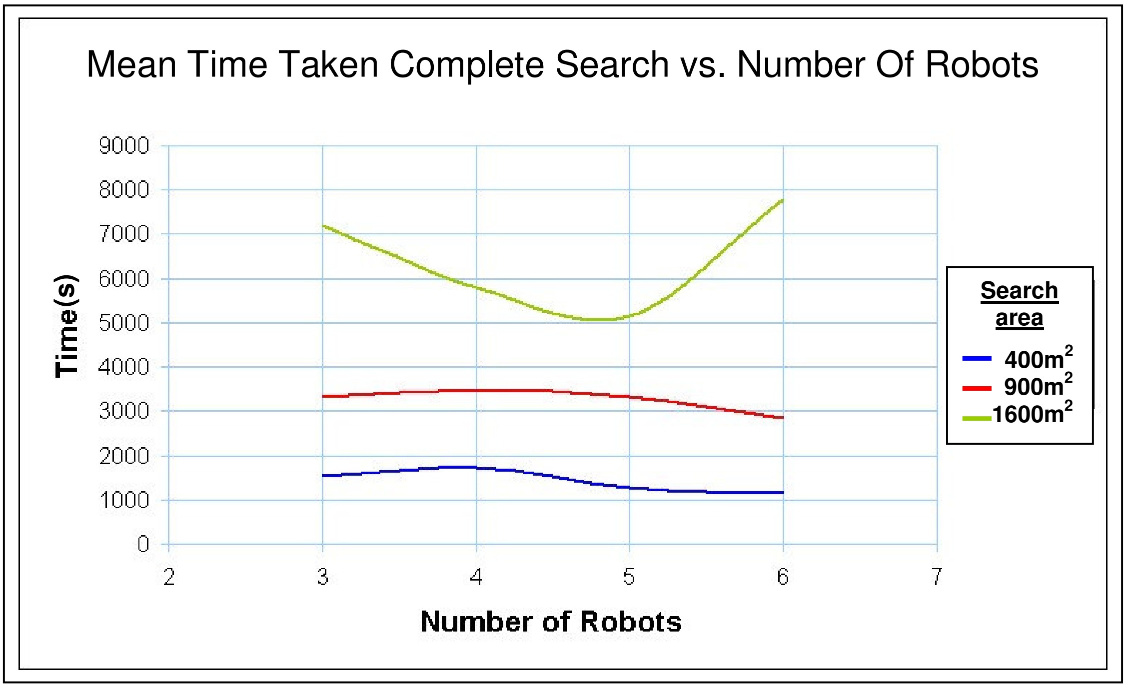


Figure 7-2

Graph of the mean time taken to locate target versus number of robots.

in completing a search for an additional robot) was observed when the number of robots was further increased to the six.

We believe that for the larger search area, the significant changes in times taken were due to the change in the aggregate speed of the multiple robots (the speed of the centre of mass of the robots) when the number of robots in it was varied. In order to substantiate this, we have done a further study on the speed of different size multiple robot system moving in the cluttered environment using our algorithm.

Similarly, different size multiple robot systems were started with each robot 0.5m apart at different positions and directions along the periphery of the environment, but then, instead of the time taken to complete the search, the time taken and the displacement of the centre of mass of the robots from their initial positions to the opposite periphery were recorded. And using the information, the average aggregate speed the multiple robots was computed after fifty simulation runs for each configuration and plotted on a graph shown on Figure 7-3.

From this plot, it was observed that the average speed of our multiple robot system decreased as the number of robots increased (approximately a decrease of 0.004m/s for an additional robot). And we believe that the flocking of the robots was the reason for the slowing down of the robot system.

When the number of robots increased, flocking would cause some robots to

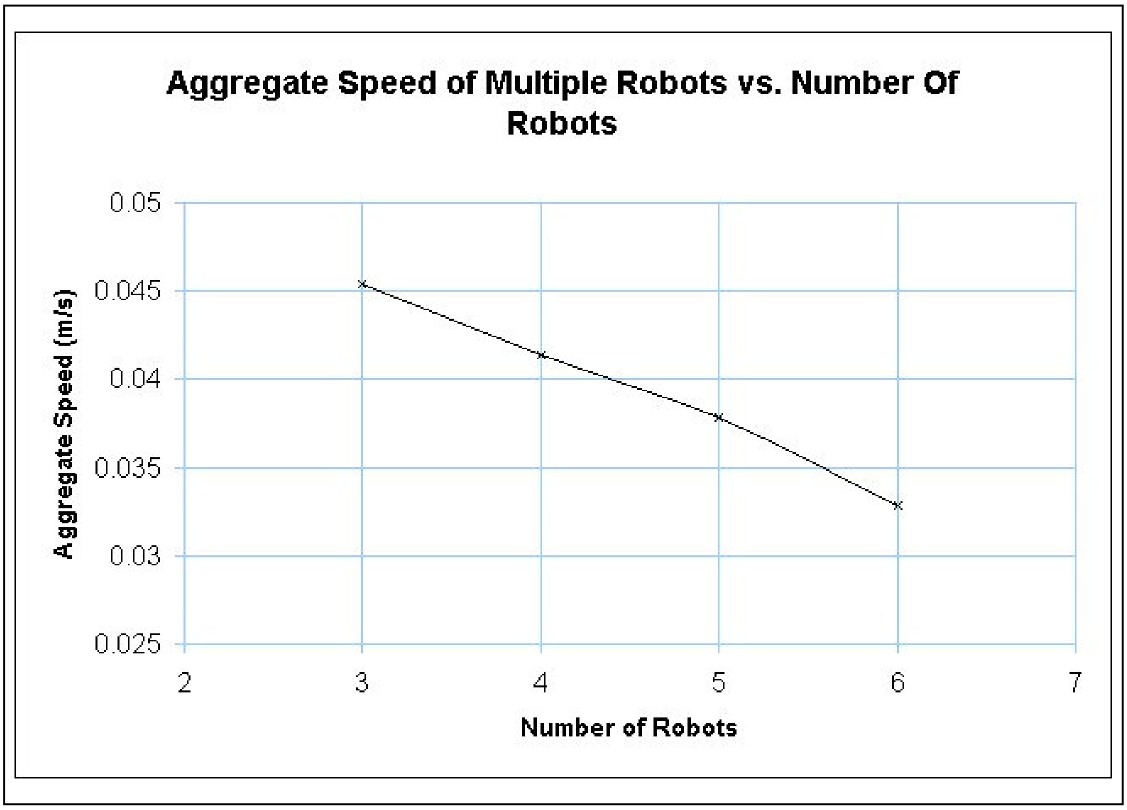


Figure 7-3

Graph of aggregate speed of multiple robots versus the number of robots.

become obstacles to other robots. And this became more prominent, when the number of robots increased. As a result, the multiple robots would have to spend additional time negotiating other robots, and hence slowing down the aggregate speed of the whole system.

Referring back to the plot of the time to complete the search versus the number of robots in Figure 7-2, when search area was relatively smaller, the difference in speed and number of robots had not caused significant changes in the time taken to complete the search. However for a bigger search area of 1600m^2, an increase in the number of robots from three to five had in fact managed to cause a reduction in the time taken despite the decrease in speed. This implied that the size of the multiple robot system was most probably the determining factor affecting the time taken then. In contrast, as the number of robots was further increased to six, the time taken began to increase. This was probably because the decrease in aggregate speed of the multiple robot system had become unfavourable enough to cause the whole search process to slow down despite having more robots.

7.4 EXPERIMENTS FOR DIFFERENT SIZE SEARCH AREA

7.4.1 Procedure

For this set of simulations, the procedure was similar to the previous, just that the time taken for searching different size search area was the issue of concern. In the simulation, the multiple robot system was deployed for search area of 400m^2, 900m^2 and 1600m^2.

Similarly the number of robots was varied from three to six for each search area and the robots were also initially placed 0.5m apart at the periphery of the search area and heading in the same direction into the search area. These initial positions and directions, and the position of the target were changed for every new simulation run and the mean times taken to complete the searches were obtained.

7.4.2 Results Evaluations

Figure 7-4 shows the plot of the results obtained. It depicts the relationship

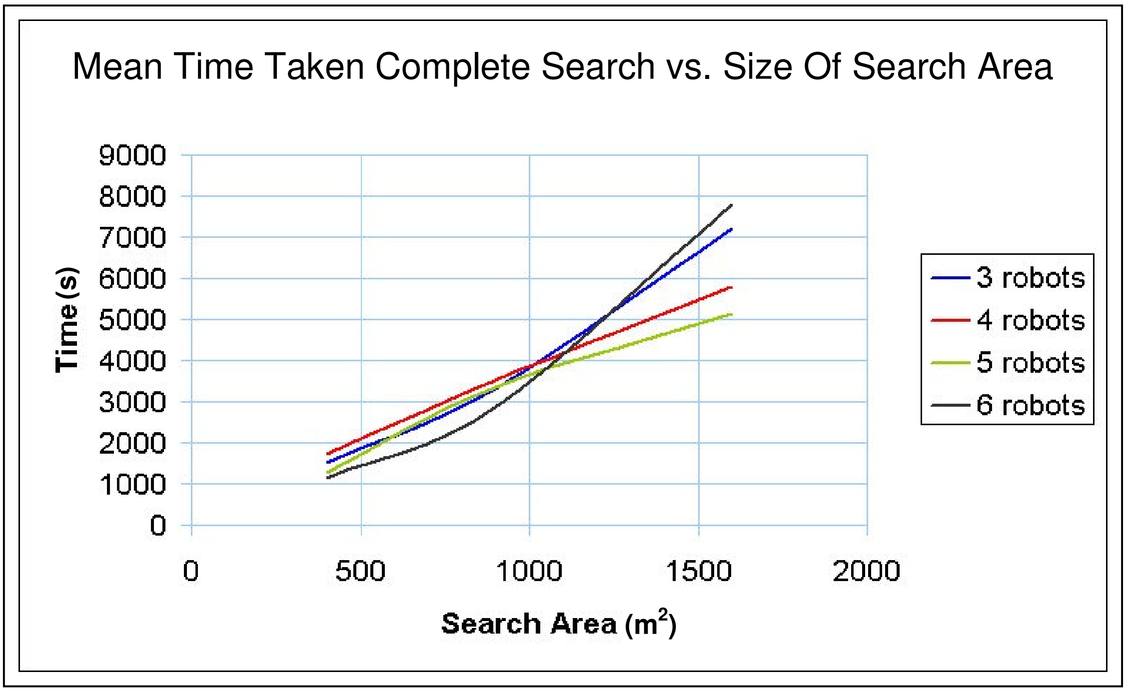


Figure 7-4

Graph of the mean time taken to locate target versus search area.

between the time taken to complete the search and different size of search area.

It was observed from the graph that for search area of around 400m^2, the time taken by multiple robot system of different robots were almost the same, with multiple robot system of 4 robots taking the longest time followed by 3 robots, 5 robots and finally 6 robots. And as expected, the curves have positive gradients since an increase in the size of search area should result in the corresponding increase in the time taken to accomplish the search. However, closer examination showed that as the search area increase, the difference in time taken became more obvious, and the sequence of time taken also began to change. At the point when the search area was increased to 1600m^2, a multiple robot system with 6 robots had actually taken the longest time followed by 3 robots, 4 robots and 5 robots.

Similar to previous section, the time taken by 6 multiple robots in a 1600m^2 search area was unforeseen. This was because we were expecting a multiple robot system composing 6 robots to take the shortest time to complete the search, but the results showed otherwise. We believed that the reason for this was again due to the slowing down of system when number of robots increased. For the larger search area, the advantage of having more robots was only realized from 3 robots to 5 robots, but when the number of robots was increased to 6, the slowing down of the multiple robot system became prominent enough to cause longer time to complete the search.

7.5 CHAPTER SUMMARY

In this chapter we have presented the experiments we carried out to gauge the performance of our search algorithm. The results showed that when the search area was relatively small, increasing the number of robots from three to six did not lead to significant difference in the time taken to locate the target. One the other hand, for a larger search area $1600m^2$, the increase in number of robots from three to five had caused relatively notable reduction in the time taken, but a further increase to six robots had actually caused the time taken to increase. We believed that this increase in time taken had been caused by the reduction in speed of the multiple robot system when more robots were present. An experiment was done to verify this reduction in speed for more robots, and the results had proven the hypothesis true.

Chapter 8

CONCLUSION

8.1 THESIS CONCLUSION

For this project, we have formulated an algorithm that has coordinated the movements of multiple robots to follow a search tactic collectively in an unknown and cluttered environment.

Firstly we have designed the system architecture for the multiple robots. This architecture is one that is decentralized, autonomous, localized and homogeneous.

Next we developed the individual robot reactive behaviour that will make their coordinated movement possible. The algorithm is to program every robot with the same set of primitive behaviours: (1) obstacles negotiation, (2) homing, (3) flocking and (4) migration, with obstacles negotiation being the most important to migration being the least important. Obstacles negotiation will enable a robot to choose an obstacles-free path to move; homing will guide the robots towards the target upon detection; flocking will ensure the robots move in a group and migration helps a robot displaced in the specified search direction. According to different environment stimulants, the robots will adopt one of these behaviours at any one time according to their order of importance. And in order to justify these behaviours' feasibility, we have carried out simulation and hardware tests to make sure they worked. Therefore we have also built robots that provided us with the physical platform for implementing our algorithm.

Thirdly we have gone on to conduct multiple simulations to gauge our algorithm's performance. For this, we have conducted the multiple simulations to find out how the time taken to complete the search would vary for different number of robots used and different size environment that required to be searched. The results showed that for smaller search area, the time taken for different number of robots did not vary significantly. However when the search areas were increase to 1600m^2, an increase in the number of robots from three to five resulted in shorter time taken to complete the search and when the quantity was further increased to six, the time taken actually increase. We believed that this was due to the slowing down of the aggregate speed of the multiple robot system because as the number of units increased, other than negotiating obstacles, additional time was taken to negotiate the nearby robots. We have carried out another set of simulations to justify this decrease in speed.

8.2 POSSIBLE FUTURE WORKS

1. Further studies can be conducted to examine the influence and effectiveness of communication among the robots. There may be more information that can be passed so that the time taken to complete the search can further decrease.
2. As the current problem only dealt with cluttered environment, another possible future work can be formulating an algorithm that can also work in structured environment. This may imply that the robots may need to be able to distinguish cluttered and structured obstacles so that they can perform the suitable obstacles negotiation behaviours.
3. (Kube and Bonabeau, 2000) pointed out that a common problem of multiple robot system was stagnation. This was because without global knowledge, a group of robots might find itself in a deadlock where it could not make any progress. Therefore study may be made on implementing a partially autonomous system instead so that in any case of deadlocking or other unexpected contingencies, human intervention can come in to try to resolve the situation.
4. Another future work that can be carried out is the study on how the performance of a search will be like when there is more than one group of multiple robots searching inside the search environment.

Bibliography

[1] Alami, R., Bothelho, S. S. C.: ‘Plan-Based Multi-robot Cooperation’, M. Beetz *et al* (Eds.): Plan-Based Control of Robotic Agents, LNAI 2466, pp. 1 – 20 (2002)

[2] Arkin, R. C., Balch, T.: ’Cooperative Multiagent Robotic Systems’, Mobile Robot Laboratory, College of Computing, Georgia Institute of Technology (1997)

[3] Arkin, R. C.: ‘Motor Schema-Based Mobile Robot Navigation’, The International Journal of Robotics Research, pp. 92 – 122 (1989)

[4] Arkin, R. C.: ’Cooperation without communication: ‘Multiagent schema based robot navigation’, Journal of Robotics Systems 9(3), pp. 351 – 364 (1992)

[5] Asama, H., Ozaiki, K., Itakura, H., Matsumoto, A., Ishida, Y., Endo, I: ‘Collision avoidance among multiple mobile robots based on rules and communication’, International Robots and System (IROS) Nov, pp. 1215 – 1220 (1991)

[6] Balchen, J. G.: ’Modelling of fish behaviour’, Predictability and Nonlinear Modelling in Natural Sciences and Economics, pp. 282 – 293 (1994)

[7] Beni, G., Hackwood, S., Liu, X.,: ‘High-order Strictly Local Swarms’, International Conference of Distributed Robotic and Automation Systems, Osaka, Japan, pp. 267 – 278 (1994)

[8] Beni, G.: ’The Concept Of Cellular Robotic System’, IEEE International Symposium on Intelligence Control, pp. 57 – 62 (1988)

[9] Bonabeau, E. and Théraulaz, G.: ’Swarm Smarts’, Scientific American, March, pp 73 – 79 (2000)

[10] Borenstein, J., Koren, Y.: ’Real-time obstacle avoidance for fast mobile robots in cluttered environments’, Proceedings of IEEE International Conference on Robotics and Automation, pp. 572 – 577 (1990)

[11] Camazine, S., Deneubourg, J. -L., Franks, N. R., Sneyd, J., Théraulaz, G., Bonabeau, E.: ‘Self-Organization in Biological Systems’, Princeton University Press (2001)

[12] Cao, Y. U., Fukunaga, A. S., Kahng, A. B.: ’Cooperative mobile robotics: Antecedents and directions’, Autonomous Robots 4(1), pp. 7 – 27 (1995)

[13] Gage, D. W.: ‘Randomized search strategies with imperfect sensors’, Preceedings of SPIE Mobile Robots VIII, Boston, 9 – 10 September, Vol 2058, pp. 270 – 279 (1993)

[14] Gage, D. W.: ’Command Control For Many-Robot Systems’, Proceedings of AUVS92, Huntsville AL, 22 – 24 June, pp. 28 – 34 (1992)

[15] Gelenbe, E., Schmajuk, N., Staddon, J., Reif, J.: ‘Autonomous search by robots and animals: A Survey’, Robotics and Autonomous Systems 22, pp. 23 – 34 (1997)

[16] Gerkey, B. P., Matarić, M. J.: ‘Sold!: Auction methods for multirobot coordination’, IEEE Transactions on Robotics and Automation 18(5), pp. 758 – 767 (2002)

[17] Koren, Y., Borenstein, J.: ‘Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation’, Proceeding of the IEEE Conference on Robotics and Automation, Sacramento, California, April 7 – 12, pp. 1398 – 1404 (1991)

[18] Kube, C. R and Bonabeau, E: ‘Cooperative transport by ants and robots’, Robotics and Autonomous Systems 30, pp. 85 – 101 (2000)

[19] Kube, C. R. and Zhang, H.: ‘Collective Robotic Intelligence’, Second International Conference on Simulation of Adaptive Behavior’, pp. 160 – 168 (1992)

[20] LaValle, S., Lin, D., Guibas, L. J., Latombe, J. –C., Motwani, R.: ‘Finding an Unpredictable Target in Workspace with Obstacles’, Proceedings of IEEE International Conference on Robotics and Automation, pp. 1 – 6 (1997)

[21] Lincoln, F. C.: ‘Migration of Birds’, U.S. Fish and Wildlife Service, Circular 16 (1935), revised by Steven R. Peterson (1979), revised by John L. Zimmerman, Division of Biology, Kansas State University, Manhattan KS (1998). http://www.npwrc.usgs.gov/resource/othrdata/migratio/migratio.htm

[22] Marrow, P., Ghanea-Hercock, R.: ‘Mobile software agents – insect-inspired computing’, BT Technol J Vol 18, pp. 129 – 138 (2000)

[23] Matarić, M. J., Sukhatme, G. S., Ostergaard, E. H.: ’Multi-robot task allocation in uncertain environments’, Autonomous Robots 14, pp. 255 – 263 (2003)

[24] Musser, G.: ‘Robots That Suck’, Scientific American, February, pp. 72 – 73 (2003)

[25] Park, D. J., Mullins, B. E.: ’Toward Finding an Universal Search Algorithm for Swarm Robots’, Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, pp. 1945 – 1950, October (2003)

[26] Parker, L. E.: ‘Distributed Algorithms for Multi-Robot Observation of Multiple Moving Targets’, Autonomous Robots 12, pp. 231 – 255 (2002)

[27] Parker, L. E.: ’Current Research In Multi-Robot Systems’, Journal of Artificial Life and Robotics 7 (to appear) (2003)

[28] Payton, D., Daily, M., Estowski, R., Howard, M., Lee, C.: ’Pheromone Robotics’, Autonomous Robots 11, pp. 319 – 324 (2001)

[29] Payton, D., Estkowski, R., Howard, M.: ’Progress in Pheromone Robotics’, Intelligent Autonomous Systems 7, pp. 256 – 264 (2002)

[30] Reynolds, C. W.: ’Flocks, Herds, and Schools: A Distributed Behavioural Model’, Computer Graphics 21(4), pp. 25 – 34 (1987)

[31] SPAWAR (Space and Naval Warfare) Systems Center San Diego: ’Many-Robot Systems’,

http://www.spawar.navy.mil/robots/research/manyrobo/manyrobo.html (Last updated: 1998)

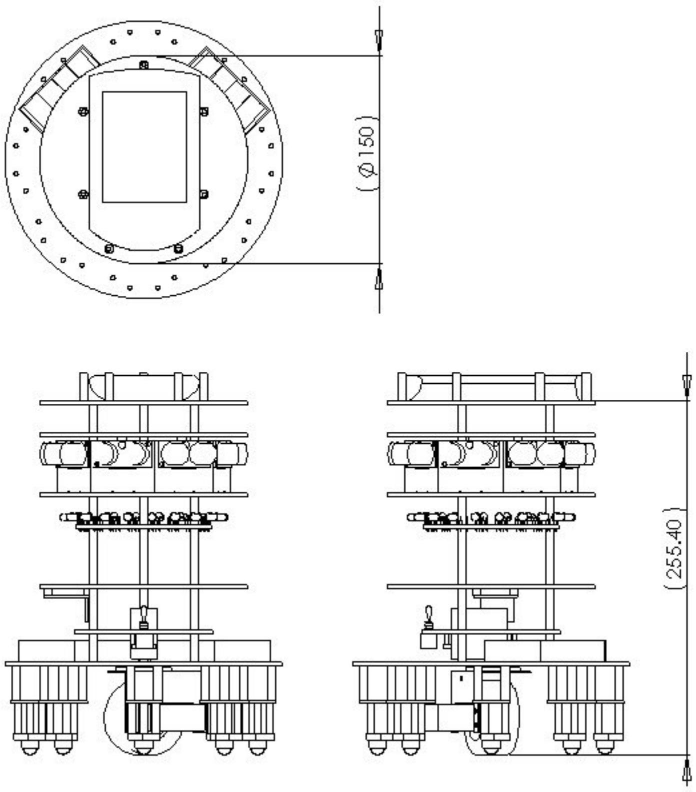
[32] Sudd, J. H., Franks, N. R.: In ’The Behavioural Ecology of Ants’, Glasgow: Blackie , New York: Chapman and Hall, pp. 98 – 119 (1987)

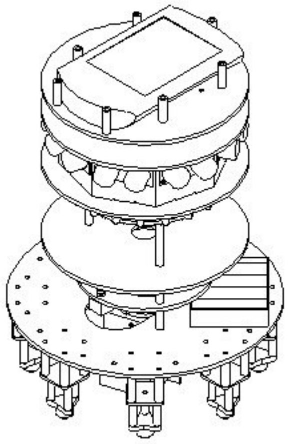
[33] Wagner, I. A., Lindenbaum, M., Bruckstein, A. M.: ’Distributed Covering By Ant-Robots Using Evaporating Traces’, IEEE Transactions on Robotics and Automation, Vol. 15, No. 5, October (1999)

[34] Yamashita, A., Ota, J., Asama, H.; ‘Motion Planning of Multiple Mobile Robots for Cooperative Manipulation and Transportation’, IEEE Transactions on Robotics and Automation, vol 19, No. 2, pp. 223 – 237 (2003)

APPENDICES

CAD DRAWING OF ROBOT





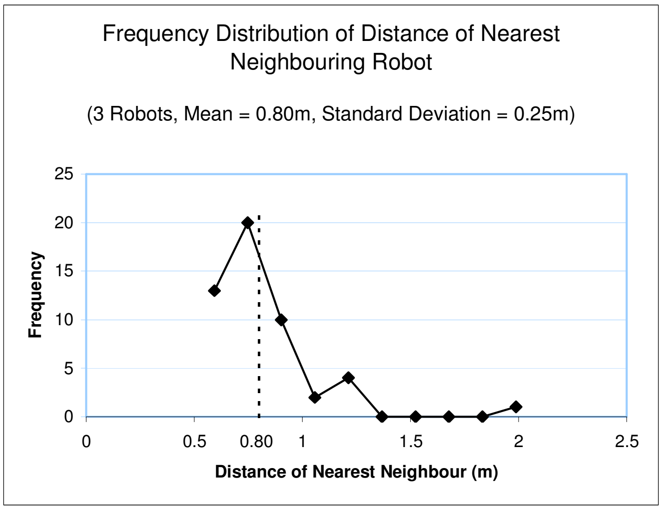
FLOCKING BEHAVIOUR EXPERIMENT DATA (CHAPTER 4)

3 Robots

S/no.	Distance From Nearest Robot
1	0.92137
2	0.840428
3	0.756427
4	0.730072
5	0.520872
6	0.795372
7	0.858072
8	2.055346
9	0.730593
10	0.588938
11	0.677422
12	0.754834
13	0.837471
14	0.683815
15	0.888224
16	0.842108
17	0.593834
18	0.817279
19	1.129983
20	1.123864
21	0.589765
22	0.921178
23	0.730403
24	0.616252
25	0.890524
26	0.668403
27	0.708104
28	0.678977
29	0.715114
30	0.748771
31	1.267464
32	0.884531
33	0.636356
34	0.580637
35	1.241001
36	0.78398
37	0.681217
38	0.616232
39	0.569429
40	0.693699
41	1.018577
42	1.015078
43	0.697204
44	0.59923
45	0.660734
46	0.963761
47	0.581517
48	0.698986
49	0.567272
50	0.704432
Mean	0.79750304

Number of Categories	10
Min	0.52
Max	2.06
Category length	0.155

Category	Lower Bound	Uper Bound	Mean Value	Frequency
1	0.515	0.67	0.5925	13
2	0.67	0.825	0.7475	20
3	0.825	0.98	0.9025	10
4	0.98	1.135	1.0575	2
5	1.135	1.29	1.2125	4
6	1.29	1.445	1.3675	0
7	1.445	1.6	1.5225	0
8	1.6	1.755	1.6775	0
9	1.755	1.91	1.8325	0
10	1.91	2.065	1.9875	1
Total				50


Frequency Distribution of Distance of Nearest Neighbouring Robot

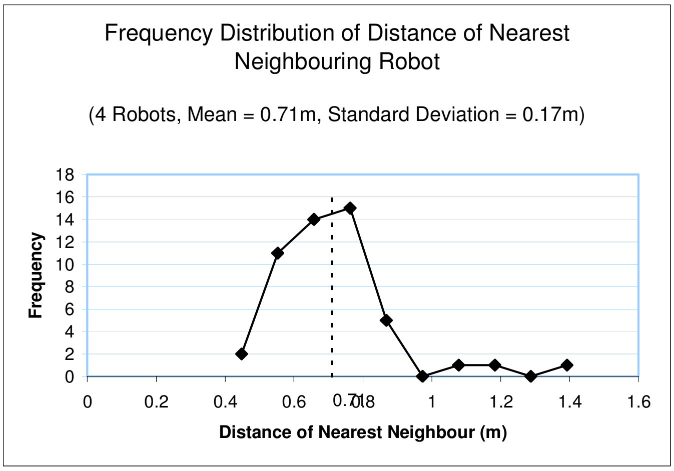
(3 Robots, Mean = 0.80m, Standard Deviation = 0.25m)

4 Robots

S/no.	Distance From Nearest Robot
1	0.771763
2	1.053437
3	0.769312
4	0.882340
5	0.600510
6	0.553444
7	0.640475
8	0.523434
9	0.740939
10	0.731699
11	0.805617
12	0.572972
13	0.913297
14	0.553862
15	0.715840
16	0.691290
17	0.767000
18	0.674052
19	0.860481
20	0.692318
21	0.704060
22	0.665237
23	0.609110
24	0.749388
25	0.606856
26	0.633321
27	0.547027
28	0.579012
29	0.593837
30	0.759843
31	0.850965
32	0.447626
33	0.612639
34	1.142769
35	0.649489
36	0.823044
37	0.615411
38	0.803768
39	0.596726
40	0.728494
41	0.599862
42	1.444847
43	0.662313
44	0.659953
45	0.780490
46	0.599477
47	0.710689
48	0.528344
49	0.661332
50	0.400203
Mean	0.705604

Number of Categories	10
Min	0.40
Max	1.44
Category length	0.105

Category	Lower Bound	Uper Bound	Mean Value	Frequency
1	0.395	0.5	0.4475	2
2	0.5	0.605	0.5525	11
3	0.605	0.71	0.6575	14
4	0.71	0.815	0.7625	15
5	0.815	0.92	0.8675	5
6	0.92	1.025	0.9725	0
7	1.025	1.13	1.0775	1
8	1.13	1.235	1.1825	1
9	1.235	1.34	1.2875	0
10	1.34	1.445	1.3925	1
Total				50


Frequency Distribution of Distance of Nearest Neighbouring Robot

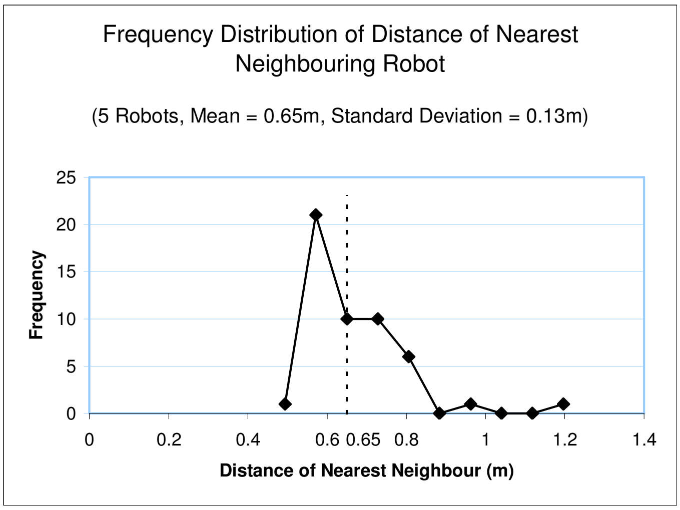
(4 Robots, Mean = 0.71m, Standard Deviation = 0.17m)

5 Robots

S/no.	Distance From Nearest Robot
1	0.617077
2	1.288127
3	0.778022
4	0.644263
5	0.767255
6	0.595421
7	0.812607
8	0.771261
9	0.614141
10	0.464041
11	0.698635
12	0.647987
13	0.65635
14	0.930403
15	0.715641
16	0.632361
17	0.786234
18	0.60004
19	0.731598
20	0.476579
21	0.71296
22	0.571321
23	0.504817
24	0.565228
25	0.726202
26	0.576157
27	0.551291
28	0.52293
29	0.583427
30	0.77407
31	0.686227
32	0.513562
33	0.663038
34	0.594936
35	0.585726
36	0.675441
37	0.719058
38	0.737171
39	0.577482
40	0.757329
41	0.522282
42	0.48107
43	0.573219
44	0.533055
45	0.651426
46	0.59648
47	0.600268
48	0.727847
49	0.649683
50	0.585503
Mean	0.654945

Number of Categories	10
Min	0.46
Max	1.23
Category length	0.078

Category	Lower Bound	Uper Bound	Mean Value	Frequency
1	0.455	0.533	0.494	1
2	0.533	0.611	0.572	21
3	0.611	0.689	0.65	10
4	0.689	0.767	0.728	10
5	0.767	0.845	0.806	6
6	0.845	0.923	0.884	0
7	0.923	1.001	0.962	1
8	1.001	1.079	1.04	0
9	1.079	1.157	1.118	0
10	1.157	1.235	1.196	1
Total				50


Frequency Distribution of Distance of Nearest Neighbouring Robot

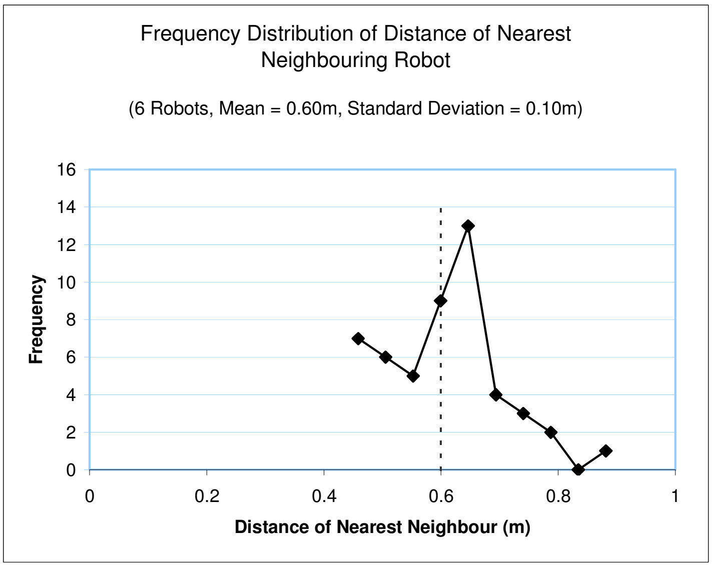
(5 Robots, Mean = 0.65m, Standard Deviation = 0.13m)

6 Robots

S/no.	Distance From Nearest Robot
1	0.721039
2	0.620484
3	0.580714
4	0.676291
5	0.473432
6	0.678053
7	0.673144
8	0.5349
9	0.563765
10	0.453652
11	0.725026
12	0.902567
13	0.65959
14	0.573542
15	0.519925
16	0.587345
17	0.495088
18	0.591682
19	0.791111
20	0.665083
21	0.658481
22	0.790445
23	0.438048
24	0.614856
25	0.470337
26	0.620352
27	0.648476
28	0.621146
29	0.63684
30	0.545068
31	0.612472
32	0.486692
33	0.495574
34	0.521989
35	0.633124
36	0.564994
37	0.517537
38	0.630312
39	0.606145
40	0.468989
41	0.621499
42	0.656023
43	0.636135
44	0.455937
45	0.632306
46	0.589558
47	0.458551
48	0.731707
49	0.617626
50	0.560153
Mean	0.600556

Number of Categories	10
Min	0.44
Max	0.90
Category length	0.047

Category	Lower Bound	Uper Bound	Mean Value	Frequency
1	0.435	0.482	0.4585	7
2	0.482	0.529	0.5055	6
3	0.529	0.576	0.5525	5
4	0.576	0.623	0.5995	9
5	0.623	0.67	0.6465	13
6	0.67	0.717	0.6935	4
7	0.717	0.764	0.7405	3
8	0.764	0.811	0.7875	2
9	0.811	0.858	0.8345	0
10	0.858	0.905	0.8815	1
Total				50



SIMULATION FOR VALIDATION EXPERIMENT (CHAPTER 6)

3 Robots

S/no.	Time
1	169.53
2	562.62
3	1060.77
4	3391.08
5	3114.99
6	1313.73
7	1946.07
8	1925.94
9	375.27
10	1595.04
11	4382.58
12	261.36
13	387.45
14	1522.92
15	770.34
16	1684.59
17	1622.22
18	156.36
19	209.40
20	1873.14
21	1329.21
22	2013.54
23	1114.86
24	5264.43
25	2990.64
26	197.85
27	2355.81
28	253.92
29	523.32
30	1162.83
31	81.84
32	90.87
33	558.09
34	1032.81
35	303.84
36	18280.29
37	950.01
38	213.45
39	3716.43
40	444.72
41	843.87
42	675.21
43	3660.42
44	3469.38
45	212.46
46	435.39
47	1473.39
48	470.91
49	936.57
50	11068.86
51	441.30
52	413.88
53	164.70
54	1546.08
55	988.08
56	2199.78
57	649.95
58	172.77
59	2009.22
60	672.00
61	474.99
62	311.58
63	141.87
64	1018.80
65	114.27
66	1841.07
67	352.41
68	140.04
69	1921.32
70	2714.70
71	543.66
72	141.33
73	128.01
74	151.83
75	1828.71
76	5952.27
77	88.59
78	2455.62
79	179.40
80	145.38
81	794.13
82	850.80
83	1984.50
84	315.87
85	632.67
86	12417.09
87	2197.29
88	1777.17
89	73.14
90	1746.93
91	1226.01
92	361.59
93	149.40
94	1041.00
95	3110.67
96	1678.29
97	1051.59
98	199.86
99	2773.83
100	547.77
101	384.36
102	913.77
103	936.48
104	967.35
105	911.58
106	511.20
107	1256.01
108	245.37
109	446.37
110	1175.28
111	520.41
112	2596.47
113	3463.14
114	452.64
115	452.43
116	110.01
117	138.69
118	136.56
119	1910.34
120	723.06
121	718.23
122	320.13
123	880.53
124	430.98
125	633.93
126	867.84
127	3065.07
128	3039.51
129	761.01
130	636.54
131	3196.08
132	410.94
133	103.98
134	107.73
135	840.24
136	572.61
137	2136.45
138	1718.64
139	5411.13
140	2604.78
141	1470.24
142	2484.00
143	249.63
144	3668.82
145	1796.61
146	751.17
147	1336.92
148	11577.18
149	353.13
150	1775.94
151	1312.59
152	198.99
153	775.08
154	15840.33
155	242.61
156	1525.11
157	1985.31
158	440.67
159	2097.39
160	275.01
161	1382.55
162	1732.50
163	505.17
164	2869.41
165	174.93
166	194.79
167	1141.17
168	1128.21
169	2159.13
170	105.27
171	92.97
172	100.98
173	504.03
174	2738.58
175	1359.30
176	60.84
177	23.37
178	34.65
179	44.58
180	22.62
181	24.84
182	26.16
183	24.69
184	19.98
185	35.64
186	32.76
187	42.69
188	85.05
189	50.25
190	724.23

Mean	1439.95
Standard Deviation	2366.9

4 Robots

S/no.	Time
1	174.99
2	248.85
3	2103.12
4	2085.39
5	612.69
6	248.16
7	1362.39
8	150.54
9	263.85
10	169.53
11	407.04
12	3640.92
13	3227.94
14	556.02
15	842.73
16	1416.09
17	472.83
18	3161.82
19	85.23
20	181.92
21	133.86
22	487.41
23	1411.68
24	84.54
25	46.83
26	76.77
27	90.69
28	1879.32
29	17351.73
30	100.74
31	1843.20
32	13.26
33	16.77
34	45.57
35	42.78
36	37.86
37	55.74
38	756.84
39	73.86
40	1093.11
41	142.74
42	123.57
43	1748.25
44	926.43
45	620.37
46	269.13
47	221.91
48	293.52
49	1494.81
50	416.70
51	2498.70
52	1942.32
53	1361.10
54	1240.68
55	380.04
56	1420.02
57	963.66
58	1671.87
59	285.81
60	602.28
61	6293.91
62	829.83
63	383.58
64	1890.87
65	232.53
66	1189.29
67	721.08
68	2103.06
69	3433.44
70	865.08
71	2722.98
72	196.62
73	2204.28
74	2398.74
75	644.46
76	197.25
77	202.23
78	845.07
79	1046.22
80	733.95
81	760.32
82	300.42
83	220.56
84	319.44
85	258.90
86	694.23
87	2633.07
88	949.02
89	1293.15
90	378.39
91	292.68
92	365.64
93	286.38
94	189.63
95	720.30
96	2028.51
97	811.11
98	1260.00
99	2552.22
100	547.74
101	1480.41
102	519.27
103	229.05
104	124.50
105	1265.70
106	532.68
107	1014.12
108	1161.78
109	156.54
110	439.08
111	2400.06
112	2009.52
113	708.66
114	315.33
115	3599.49
116	694.83
117	1892.25
118	2868.21
119	149.55
120	1606.08
121	1004.73
122	1014.00
123	3317.34
124	143.79
125	302.88
126	2114.01
127	2528.52
128	120.00
129	100.95
130	631.86
131	190.35
132	2453.49
133	1875.24
134	86.40
135	121.08
136	550.56
137	472.11
138	164.76
139	120.39
140	151.80
141	217.17
142	1196.88
143	1424.73
144	150.93
145	1675.95
146	65.97
147	87.96
148	130.71
149	622.23
150	167.64
151	146.10
152	326.40
153	1073.82
154	237.09
155	137.73
156	140.94
157	293.37
158	652.35
159	225.33
160	956.34
161	1262.79
162	222.33
163	220.92
164	254.10
165	2460.36
166	1237.53
167	2253.90
168	2059.56
169	1851.60
170	821.10
171	161.43
172	173.19
173	382.38
174	666.99
175	1043.28
176	1275.48
177	843.30
178	477.60
179	1058.70
180	463.14
181	644.10
182	316.86
183	700.44
184	2407.77
185	336.33
186	377.04
187	688.80
188	2088.33
189	327.75
190	87.72
191	536.16
192	441.90
193	100.65
194	2746.02
195	3351.18
196	1011.78
197	106.95
198	117.72
199	615.39
200	598.26
201	275.55
202	899.28
203	158.55
204	166.35
205	1565.76
206	724.89
207	233.01
208	1117.47
209	848.28
210	1067.79
211	1764.12

Mean	985.06
Standard Deviation	1464.48

PERFORMANCE ANALYSIS (CHAPTER 7)

400m^2, 3 Robots

S/no.	Time
1	344.61
2	59.88
3	86.88
4	79.65
5	78.54
6	165.36
7	1160.46
8	656.10
9	447.33
10	1782.57
11	2487.21
12	2613.63
13	312.63
14	303.48
15	1455.90
16	2244.48
17	3940.38
18	605.28
19	420.96
20	970.47
21	1429.20
22	692.91
23	2120.40
24	1702.17
25	564.39
26	833.58
27	2565.96
28	590.22
29	5179.68
30	520.74
31	3664.32
32	1733.94
33	530.58
34	219.72
35	1406.94
36	557.91
37	386.49
38	248.49
39	175.23
40	2390.19
41	1086.24
42	827.28
43	284.49
44	3238.20
45	112.68
46	5924.10
47	4129.41
48	4175.76
49	1237.44
50	2937.09
51	3193.89
52	1510.47
53	310.95
54	1002.99
55	394.41
56	1921.29
57	338.07
58	235.92
59	4288.83
60	345.18
61	1780.83
62	1606.74
63	3212.01
64	1650.24
65	178.95
66	148.56
67	3955.68
68	1529.16
69	1039.05
70	1971.81
71	100.17
72	95.04
73	1024.59
74	2062.17
75	6632.58
76	1220.58
77	1464.99
78	2457.42
79	364.74
80	1779.69
81	1079.01
82	3157.50
83	1505.70
84	1666.44
85	5417.97
86	324.99
87	290.58
88	2110.80
89	4480.02
90	11.31
91	11.13
92	7.77
93	25.83
94	4.77
95	11.79
96	14.67
97	58.65
98	44.94
99	36.96
100	839.01
101	207.00
102	822.66
103	621.57
104	250.65
105	173.67
106	1903.29
107	3718.53
108	4494.51
109	2827.26
110	366.69
111	304.56
112	328.74
113	401.16
114	4010.88
115	588.51
116	2091.72
117	381.75
118	3042.21
119	804.42
120	1989.12
121	17.31
122	1.29
123	2.67
124	309.75
125	3.06
126	2.73
127	23.61
128	14.16
129	15.48
130	28.29
131	36.48
132	59.01
133	3269.79
134	1449.99
135	310.08
136	761.25
137	2420.58
138	329.97
139	1773.36
140	1118.43
141	3209.88
142	1072.92
143	4261.77
144	960.27
145	4330.62
146	4337.16
147	857.97
148	656.22
149	446.55
150	834.24
151	3175.59
152	756.99
153	3886.83
154	598.80
155	6568.95
156	253.59
157	150.39
158	185.28
159	9575.97
160	2033.91
161	3546.12
162	220.20
163	1621.35
164	870.09
165	871.59
166	1548.39
167	2946.57
168	6977.01
169	3505.71
170	2349.21
171	1579.71
172	5050.71
173	455.70
174	1403.67
175	861.03
176	2424.99
177	2943.84
178	163.17
179	2527.56
180	1641.45
181	167.52
182	838.65
183	2004.57
184	441.69
185	3968.76
186	1461.78
187	3110.04
188	4250.52
189	282.69
190	4848.63
191	598.23
192	1700.55
193	1911.60
194	421.05
195	325.89
196	650.85
197	1658.88
198	103.53
199	972.21
200	1381.95

Mean	1523.18
Standard Deviation	1627.58

$400m^2$, 4 Robots

S/no.	Time
1	265.83
2	2656.65
3	790.41
4	5199.90
5	710.85
6	9429.63
7	2280.99
8	151.20
9	124.02
10	706.74
11	2770.89
12	531.03
13	727.83
14	1103.70
15	3928.44
16	314.64
17	3508.86
18	3273.75
19	147.24
20	2089.59
21	209.16
22	148.32
23	117.54
24	118.98
25	1418.31
26	1758.30
27	72.42
28	73.83
29	1343.79
30	1277.94
31	842.76
32	1597.23
33	111.84
34	92.01
35	1457.43
36	2920.56
37	271.11
38	386.61
39	1097.25
40	3508.26
41	5549.82
42	3384.42
43	2344.50
44	1630.17
45	381.12
46	6051.48
47	2998.17
48	1816.62
49	2980.95
50	946.44
51	153.42
52	3990.96
53	254.61
54	223.83
55	1466.28
56	954.24
57	1218.03
58	372.30
59	10815.30
60	909.75
61	542.55
62	10633.29
63	3847.59
64	4536.57
65	288.36
66	253.32
67	2932.11
68	98.31
69	45.90
70	42.24
71	40.89
72	35.88
73	38.52
74	24.39
75	19.77
76	14.46
77	8.43
78	13.50
79	2051.19
80	3553.26
81	4180.14
82	2241.30
83	794.13
84	1026.48
85	1762.14
86	505.32
87	402.57
88	386.94
89	349.26
90	1445.28
91	7367.88
92	41.43
93	43.77
94	54.57
95	3583.17
96	566.22
97	850.26
98	1158.30
99	2399.37
100	3781.11
101	1934.52
102	376.50
103	1346.88
104	193.53
105	159.87
106	142.32
107	1293.48
108	220.53
109	6645.84
110	826.74
111	4479.57
112	2750.67
113	1695.75
114	1182.12
115	931.41
116	5177.55
117	4009.32
118	1416.84
119	842.10
120	1110.21
121	385.71
122	945.60
123	202.14
124	152.67
125	1031.22
126	721.53
127	4177.05
128	1951.20
129	2223.60
130	5316.30
131	1973.34
132	108.90
133	550.98
134	4905.72
135	16.89
136	22.53
137	47.88
138	1826.70
139	1482.54
140	139.80
141	132.42
142	174.42
143	4174.53
144	2702.04
145	258.51
146	2512.92
147	2356.68
148	130.65
149	3287.64
150	1361.22
151	74.01
152	96.27
153	6909.96
154	1517.70
155	2602.89
156	791.64
157	5414.76
158	2456.04
159	198.63
160	159.81
161	395.73
162	4705.35
163	500.97
164	217.47
165	160.53
166	118.83
167	6008.46
168	195.51
169	1077.15
170	421.68
171	1269.12
172	283.14
173	2935.38
174	7629.27
175	1219.11
176	1850.91
177	2340.09
178	216.09
179	822.39
180	3902.61
181	1166.37
182	879.99
183	2924.04
184	69.39
185	4030.98
186	138.93
187	3113.73
188	1640.82
189	2801.79
190	4748.94
191	1262.04
192	518.46
193	1740.87
194	2971.17
195	3651.90
196	1200.48
197	1151.64
198	1005.96
199	62.31
200	2680.47

Mean	1731.00
Standlard Deviation	1986.69

400m^2, 5 Robots

S/no.	Time
1	363.63
2	2788.38
3	4750.92
4	1045.77
5	5485.74
6	7215.96
7	393.21
8	722.79
9	2493.33
10	2461.50
11	403.32
12	1318.59
13	131.19
14	193.41
15	110.01
16	135.03
17	100.08
18	101.13
19	1341.54
20	5717.52
21	1946.52
22	17.70
23	41.46
24	42.69
25	1.20
26	23.25
27	1.02
28	0.03
29	25.08
30	2.10
31	4.92
32	2.25
33	8.64
34	11.28
35	6.93
36	3774.00
37	0.18
38	22.53
39	9.72
40	11.70
41	10.44
42	13.14
43	18.42
44	1393.83
45	1825.08
46	1300.26
47	2364.75
48	358.89
49	2758.17
50	3307.08
51	478.92
52	554.49
53	3244.98
54	1110.18
55	1440.39
56	2381.94
57	2791.05
58	617.46
59	3574.74
60	3758.04
61	2011.35
62	895.26
63	3189.99
64	2416.65
65	0.69
66	10.26
67	5.04
68	7.44
69	10.08
70	10.29
71	16.17
72	3146.82
73	274.86
74	1803.96
75	7843.83
76	943.23
77	1414.65
78	1343.91
79	1023.93
80	3124.59
81	4336.02
82	1574.97
83	2304.72
84	2224.11
85	472.47
86	1123.83
87	2117.67
88	1165.65
89	2865.36
90	8858.52
91	142.77
92	96.57
93	87.99
94	94.32
95	887.91
96	368.07
97	1013.82
98	957.96
99	166.89
100	191.85
101	164.13
102	3396.48
103	3959.94
104	822.00
105	904.11
106	805.89
107	1302.96
108	296.49
109	2936.85
110	66.69
111	111.18
112	53.94
113	131.43
114	22.59
115	17.67
116	9.57
117	12.00
118	8.97
119	14.37
120	10.32
121	25.35
122	10.08
123	13.41
124	11.40
125	21.45
126	9.81
127	3.84
128	3.99
129	5.37
130	1.68
131	1.77
132	10.14
133	10.98
134	8.61
135	9.60
136	8.70
137	5.37
138	1210.53
139	3306.42
140	5433.33
141	458.55
142	1811.22
143	3196.59
144	2413.71
145	1388.10
146	1391.58
147	2496.54
148	720.99
149	4251.33
150	1866.54
151	5008.92
152	1271.07
153	2142.15
154	873.45
155	4635.81
156	1639.41
157	5856.18
158	111.66
159	2668.08
160	4751.52
161	1108.26
162	1099.08
163	519.96
164	351.39
165	712.08
166	1186.68
167	2568.99
168	1150.56
169	706.32
170	1324.98
171	833.70
172	2680.32
173	3439.02
174	410.04
175	4482.27
176	482.76
177	1665.03
178	1833.27
179	4090.89
180	2.07
181	2.01
182	12.12
183	98.22
184	42.30
185	59.85
186	28.05
187	41.64
188	48.33
189	42.03
190	12.93
191	10.74
192	15.93
193	2.85
194	3617.34
195	700.62
196	1212.39
197	1123.41
198	5289.00
199	1987.53
200	1571.52

Mean	1288.76
Standard Deviation	1669.54

$400m^2$, 6 Robots

S/no.	Time
1	318.48
2	1685.88
3	3076.26
4	387.57
5	3191.13
6	240.72
7	260.94
8	250.38
9	202.71
10	2602.71
11	505.83
12	1717.02
13	767.28
14	1193.64
15	231.60
16	259.68
17	146.79
18	122.88
19	3126.24
20	1676.25
21	68.67
22	356.16
23	2957.88
24	3616.62
25	343.29
26	423.09
27	255.30
28	3981.48
29	2832.15
30	3195.18
31	461.73
32	544.74
33	698.34
34	617.34
35	883.05
36	519.78
37	2465.91
38	218.01
39	185.94
40	3299.34
41	1032.63
42	1424.07
43	656.31
44	356.94
45	1536.99
46	2778.51
47	33.12
48	31.77
49	32.88
50	32.88
51	21.12
52	34.50
53	18.00
54	22.65
55	383.13
56	7.89
57	7.44
58	10.44
59	4.26
60	4.41
61	2811.39
62	1775.19
63	3782.91
64	376.59
65	52.92
66	20.97
67	96.54
68	148.14
69	56.13
70	48.24
71	51.72
72	40.08
73	74.46
74	26.97
75	19.53
76	19.56
77	8.07
78	8.82
79	9564.12
80	53.73
81	32.94
82	36.18
83	20.76
84	13.02
85	3.99
86	4.92
87	2.64
88	3.00
89	5.43
90	8.79
91	7.50
92	5.64
93	3.87
94	4.29
95	7.32
96	9.51
97	17.13
98	13.95
99	20.07
100	11.49
101	33.48
102	29.01
103	9930.57
104	1960.56
105	635.28
106	5255.64
107	1064.73
108	3351.63
109	612.51
110	3504.72
111	2549.40
112	331.29
113	3837.06
114	6225.30
115	849.99
116	3467.07
117	432.12
118	1471.29
119	2394.60
120	3405.42
121	561.39
122	599.64
123	486.36
124	759.03
125	548.88
126	892.44
127	896.43
128	4253.34
129	807.54
130	617.37
131	520.23
132	1286.61
133	1257.51
134	2512.77
135	1801.59
136	1065.90
137	1786.62
138	2197.56
139	19.62
140	35.28
141	34.47
142	68.40
143	13.95
144	0.12
145	2.88
146	2.40
147	3.30
148	2.64
149	2.97
150	26.31
151	12.18
152	11.82
153	9.21
154	7.74
155	14.22
156	5253.57
157	1455.12
158	251.43
159	400.23
160	194.04
161	170.37
162	948.75
163	922.17
164	1200.93
165	1318.59
166	1351.62
167	1356.09
168	631.53
169	417.81
170	1242.33
171	1509.54
172	1848.12
173	2166.21
174	415.86
175	302.97
176	5660.85
177	204.87
178	145.02
179	88.44
180	1663.47
181	422.91
182	1245.06
183	1166.31
184	4110.06
185	6334.77
186	3272.07
187	1675.20
188	1661.37
189	3043.92
190	2357.22
191	3102.33
192	626.22
193	842.85
194	2930.79
195	1297.80
196	4601.94
197	212.64
198	2737.35
199	877.38
200	2579.55

Mean	1141.32
Standard Deviation	1614.56

900m^2, 3 Robots

S/no.	Time
1	513.99
2	3695.49
3	7142.58
4	11058.96
5	1891.32
6	423.12
7	3487.44
8	2737.83
9	3687.45
10	5307.99
11	1391.40
12	627.00
13	1982.19
14	8755.77
15	1682.64
16	1845.75
17	7700.91
18	5955.21
19	3843.06
20	2171.07
21	8077.20
22	435.51
23	6536.10
24	17185.17
25	1934.28
26	2071.68
27	1480.53
28	2822.01
29	1096.26
30	770.55
31	16933.80
32	4109.10
33	379.26
34	272.61
35	696.33
36	5351.94
37	398.58
38	392.01
39	2096.46
40	5720.76
41	5204.01
42	3715.98
43	1325.46
44	2396.91
45	3281.70
46	75.45
47	124.05
48	11903.82
49	708.81
50	3838.14
51	2085.78
52	206.31
53	10383.15
54	1938.78
55	2621.58
56	7740.42
57	957.93
58	4735.50
59	536.04
60	5386.20
61	2275.74
62	5301.24
63	5756.73
64	7161.54
65	6808.11
66	1412.01
67	5738.34
68	1536.36
69	8118.45
70	5004.00
71	8138.61
72	2680.32
73	485.22
74	5112.96
75	1163.28
76	1618.98
77	541.38
78	931.50
79	5162.07
80	535.86
81	6489.48
82	877.44
83	857.10
84	3691.65
85	4746.42
86	5824.77
87	11694.54
88	568.59
89	1869.69
90	736.74
91	5613.00
92	206.52
93	196.80
94	225.12
95	4581.69
96	2508.51
97	1204.41
98	600.84
99	775.92
100	1691.40
101	3202.74
102	7287.96
103	322.26
104	1435.68
105	2652.57
106	1198.02
107	1025.28
108	6028.74
109	3673.29
110	9250.08
111	1506.60
112	4489.20
113	1217.40
114	1002.99
115	6610.71
116	452.34
117	443.64
118	2657.07
119	8141.07
120	5982.45
121	3269.76
122	4578.00
123	3329.91
124	1994.88
125	1317.33
126	6548.64
127	999.36
128	1786.23
129	3183.81
130	292.41
131	8460.21
132	553.26
133	525.33
134	539.43
135	2403.63
136	5503.35
137	2052.03
138	1254.03
139	3421.65
140	1167.03
141	950.85
142	7599.21
143	2724.03
144	2324.16
145	204.78
146	2987.67
147	294.18
148	8632.68
149	1103.01
150	6918.18
151	5680.98
152	940.65
153	400.86
154	336.78
155	309.99
156	1340.58
157	5936.19
158	2562.03
159	585.00
160	5070.87
161	10969.50
162	1144.80
163	1860.30
164	2701.35
165	5784.57
166	5095.95
167	5462.28
168	1202.49
169	2179.11
170	489.00
171	2259.39
172	6241.41
173	1059.96
174	3066.96
175	405.75
176	3883.83
177	462.69
178	1876.08
179	3206.37
180	318.27
181	1059.96
182	2730.93
183	3333.51
184	6297.51
185	13446.24
186	2406.81
187	1289.43
188	1015.23
189	8883.87
190	6083.64
191	10006.08
192	2275.74
193	1340.64
194	237.12
195	215.52
196	2139.69
197	5930.79
198	1815.54
199	244.92
200	1010.88

Mean	3321.84
Standard Deviation	3157.91

900m^2, 4 Robots

S/no.	Time
1	2427.81
2	2349.72
3	4707.39
4	339.87
5	410.40
6	1728.24
7	2719.23
8	3573.45
9	188.64
10	2815.23
11	243.78
12	180.60
13	7524.30
14	2198.55
15	2683.47
16	7035.99
17	3129.78
18	1965.51
19	1158.24
20	1188.78
21	8560.41
22	5702.61
23	41.73
24	50.43
25	11401.05
26	5912.34
27	3503.82
28	939.09
29	722.97
30	446.64
31	2687.28
32	541.29
33	2052.24
34	208.38
35	7968.36
36	225.66
37	1346.40
38	2471.55
39	111.24
40	46.95
41	13694.91
42	1263.51
43	552.18
44	424.59
45	282.90
46	306.18
47	1728.12
48	131.70
49	4526.25
50	3412.05
51	10503.54
52	8452.41
53	3215.85
54	6608.52
55	7071.66
56	631.32
57	5017.74
58	339.96
59	7232.49
60	3127.92
61	8439.48
62	111.36
63	4007.07
64	13186.26
65	6357.18
66	8914.05
67	764.28
68	819.81
69	1387.26
70	496.71
71	5834.85
72	5048.46
73	1225.29
74	1054.05
75	4631.01
76	1854.66
77	2203.38
78	623.01
79	3823.98
80	1269.15
81	1729.59
82	954.84
83	11295.48
84	1013.64
85	3358.11
86	2375.13
87	3789.42
88	1423.86
89	13969.11
90	564.60
91	1243.83
92	4348.95
93	1644.72
94	652.50
95	1675.74
96	5728.68
97	2013.36
98	5398.26
99	8265.84
100	908.25
101	486.81
102	1067.76
103	851.79
104	4339.86
105	346.20
106	322.08
107	8969.10
108	383.88
109	12077.61
110	8211.00
111	464.88
112	4712.97
113	2712.30
114	3028.44
115	429.63
116	6438.90
117	6362.04
118	4349.58
119	3333.54
120	2700.87
121	5480.46
122	1696.83
123	1915.32
124	12.15
125	15810.06
126	12858.63
127	1519.92
128	1867.29
129	3068.40
130	3088.02
131	355.80
132	295.11
133	7638.15
134	3476.67
135	1148.85
136	3296.16
137	1058.10
138	5778.30
139	1740.78
140	323.76
141	2631.12
142	1918.11
143	974.04
144	2027.70
145	10860.66
146	437.94
147	1153.71
148	6714.60
149	1490.07
150	1425.90
151	1726.14
152	5708.37
153	8776.26
154	2125.62
155	3560.01
156	3326.16
157	6524.16
158	1173.48
159	1697.25
160	1942.35
161	3527.34
162	10629.90
163	1692.03
164	4117.26
165	695.28
166	6756.27
167	3680.01
168	2839.32
169	964.08
170	1462.29
171	1400.64
172	3198.96
173	1077.27
174	3943.50
175	5343.84
176	870.09
177	4205.58
178	484.26
179	12870.63
180	4360.41
181	7567.65
182	1520.10
183	2079.09
184	12269.61
185	2190.99
186	23271.66
187	5115.93
188	2310.09
189	104.43
190	129.42
191	137.64
192	6956.49
193	820.89
194	2979.33
195	3673.80
196	259.29
197	258.24
198	973.47
199	15684.72
200	2025.42

Mean	3505.49
Standard Deviation	3716.51

900m^2, 5 Robots

S/no.	Time
1	499.20
2	6968.52
3	1162.56
4	1542.39
5	4605.96
6	11400.18
7	8982.81
8	5725.14
9	637.53
10	17872.53
11	237.84
12	38.52
13	2130.57
14	13757.46
15	5890.68
16	344.19
17	417.33
18	1648.44
19	6817.89
20	5880.24
21	2383.26
22	1739.16
23	1592.49
24	6614.40
25	576.78
26	355.56
27	364.80
28	692.16
29	184.83
30	8775.72
31	210.33
32	164.55
33	190.56
34	197.16
35	12748.56
36	23.07
37	19.08
38	21.51
39	2.40
40	4068.93
41	4421.16
42	15.63
43	40.62
44	28.23
45	2618.55
46	3080.16
47	686.07
48	4797.06
49	660.03
50	3399.03
51	6516.51
52	4679.67
53	2447.16
54	5221.26
55	41.76
56	149.37
57	186.96
58	5771.10
59	307.92
60	2968.53
61	2510.79
62	643.95
63	1929.60
64	17082.48
65	9230.91
66	16086.66
67	4261.59
68	335.94
69	3478.17
70	7311.09
71	7105.35
72	5897.64
73	3376.59
74	797.52
75	3265.41
76	1286.61
77	1005.60
78	2418.99
79	3820.83
80	590.10
81	2575.95
82	2568.63
83	695.28
84	7754.43
85	9013.59
86	12951.21
87	2600.31
88	6879.96
89	3788.13
90	5358.15
91	104.31
92	3840.15
93	362.91
94	280.59
95	2407.44
96	10232.19
97	19649.52
98	3784.65
99	4302.87
100	353.37
101	2460.21
102	2409.39
103	1694.67
104	6375.18
105	448.98
106	5191.23
107	993.78
108	10632.90
109	6741.69
110	2835.78
111	1862.64
112	1386.75
113	3740.94
114	275.73
115	9767.46
116	1269.27
117	3186.60
118	2852.64
119	2921.13
120	2581.83
121	4365.00
122	4663.17
123	2580.03
124	2572.86
125	389.91
126	2920.80
127	4987.08
128	1512.18
129	209.25
130	1712.79
131	45.60
132	51.84
133	2627.52
134	4023.48
135	2407.98
136	7864.86
137	1708.47
138	2078.61
139	5395.53
140	584.43
141	4949.70
142	3159.57
143	1107.30
144	11521.50
145	2753.10
146	7243.65
147	8.43
148	5.58
149	11.79
150	16.95
151	12.30
152	2.46
153	8.76
154	4.71
155	3.72
156	0.78
157	0.09
158	1267.17
159	7466.19
160	121.56
161	52.80
162	64.26
163	70.71
164	108.33
165	68.70
166	15.54
167	14.73
168	11.34
169	13.44
170	862.65
171	11.97
172	12.99
173	11.61
174	19.47
175	23.67
176	3920.22
177	6773.91
178	1198.68
179	2920.83
180	2429.25
181	5172.18
182	4645.35
183	1215.48
184	330.09
185	1230.54
186	2794.29
187	7586.76
188	5278.83
189	1801.08
190	7410.51
191	2917.71
192	2671.71
193	3429.75
194	1673.31
195	2865.30
196	1998.57
197	2557.23
198	10659.75
199	5778.33
200	2579.88

Mean	3329.78
Standard Deviation	3669.54

900m^2, 6 Robots

S/no.	Time
1	655.20
2	1211.58
3	4244.04
4	2181.36
5	6469.08
6	2133.96
7	433.80
8	3168.48
9	399.81
10	655.20
11	1211.58
12	4244.04
13	2181.36
14	6469.08
15	2133.96
16	433.80
17	3168.48
18	399.81
19	11989.95
20	97.98
21	66.66
22	3936.42
23	210.18
24	132.33
25	118.59
26	8850.30
27	1960.71
28	4366.32
29	2338.89
30	4335.81
31	6.81
32	9.09
33	4.56
34	5.67
35	9.27
36	1.71
37	4359.75
38	632.49
39	2776.17
40	11653.32
41	565.38
42	358.77
43	7985.07
44	343.92
45	274.71
46	2625.87
47	400.86
48	226.71
49	2064.69
50	4011.51
51	2975.85
52	2540.67
53	6272.16
54	10210.71
55	1415.55
56	602.28
57	533.94
58	538.08
59	2719.80
60	298.89
61	1698.33
62	58.86
63	41.10
64	39.48
65	57.90
66	2747.46
67	2245.29
68	31.65
69	52.65
70	4414.02
71	2075.10
72	1155.66
73	10813.59
74	3528.66
75	59.91
76	83.70
77	61.80
78	304.29
79	5014.83
80	8018.19
81	2713.02
82	240.75
83	274.44
84	121.32
85	68.58
86	111.24
87	376.02
88	2795.52
89	2511.00
90	3127.32
91	333.48
92	260.85
93	3543.39
94	9188.22
95	3883.02
96	6099.51
97	381.93
98	2300.46
99	213.18
100	559.23
101	526.23
102	6776.64
103	2767.56
104	5955.33
105	698.58
106	1156.74
107	2596.41
108	7557.72
109	2299.26
110	1285.02
111	5748.75
112	1031.28
113	6386.22
114	3320.67
115	7111.26
116	5411.28
117	2804.94
118	1928.97
119	279.42
120	6299.79
121	2381.31
122	2731.83
123	590.37
124	1917.21
125	5417.28
126	2530.83
127	5059.89
128	1216.32
129	3489.57
130	3155.94
131	7633.35
132	2598.75
133	1411.95
134	6866.58
135	4584.39
136	2547.09
137	5280.66
138	1002.33
139	841.50
140	10493.94
141	1366.35
142	5823.03
143	8539.50
144	10046.70
145	9728.61
146	509.43
147	2404.02
148	3911.52
149	1639.05
150	7117.71
151	2027.34
152	3207.03
153	2354.79
154	4486.38
155	534.90
156	1488.93
157	2804.19
158	557.04
159	6340.98
160	1612.68
161	855.75
162	2286.81
163	918.27
164	3042.45
165	1941.90
166	9672.72
167	3215.49
168	2162.07
169	2646.00
170	8024.07
171	1405.56
172	2064.39
173	1215.00
174	2408.01
175	158.73
176	8397.12
177	193.05
178	535.05
179	581.46
180	2503.98
181	1087.71
182	14157.45
183	7372.50
184	9434.40
185	3455.67
186	1073.97
187	3264.48
188	4869.42
189	8057.52
190	4219.71
191	725.46
192	1351.20
193	103.23
194	80.55
195	28.32
196	35.28
197	38.58
198	153.87
199	3629.55
200	3417.99

Mean	2846.03
Standard Deviation	2938.17

1600m^2, 3 Robots

S/no.	Time
1	26902.95
2	2948.91
3	3064.86
4	9231.63
5	1107.51
6	2157.12
7	5186.79
8	332.04
9	23108.10
10	14402.28
11	2706.45
12	18596.34
13	591.30
14	8979.87
15	6951.48
16	1133.16
17	5335.29
18	10518.45
19	2872.08
20	459.87
21	4272.93
22	2748.27
23	6267.54
24	563.73
25	1067.85
26	5943.45
27	345.96
28	429.54
29	5442.51
30	3427.86
31	21678.63
32	14405.79
33	3913.05
34	10758.09
35	4484.79
36	1802.13
37	1235.34
38	1363.41
39	22469.46
40	2365.86
41	2359.86
42	9739.65
43	3921.60
44	6243.03
45	7349.34
46	599.73
47	12756.81
48	8133.69
49	2405.46
50	14253.90
51	16361.28
52	9431.10
53	22923.51
54	14418.15
55	22202.19
56	17192.07
57	4931.52
58	3271.26
59	22043.88
60	483.78
61	1207.77
62	5755.08
63	18189.09
64	4693.23
65	263.85
66	5474.07
67	3499.17
68	7384.86
69	1581.30
70	26.16
71	5693.97
72	7.38
73	10.14
74	3.45
75	5.91
76	2.37
77	7683.84
78	5566.02
79	7544.28
80	40.17
81	36.48
82	14409.54
83	9697.11
84	12874.47
85	1007.67
86	20095.26
87	9880.86
88	961.44
89	4658.82
90	12051.87
91	7432.11
92	18624.69
93	7164.00
94	18667.23
95	681.48
96	6011.55
97	12845.25
98	1592.76
99	25751.34
100	243.42

Mean	7219.14
Standard Deviation	5045.40

1600m^2, 4 Robots

S/no.	Time
1	7798.44
2	1391.61
3	828.81
4	3366.51
5	16425.00
6	9754.26
7	2552.67
8	242.28
9	87.63
10	685.20
11	508.26
12	363.03
13	82.35
14	104.49
15	86.67
16	5450.70
17	6599.04
18	1231.71
19	2677.08
20	9422.73
21	2718.96
22	9035.34
23	1076.85
24	2174.01
25	4454.04
26	9947.82
27	695.97
28	690.66
29	2530.23
30	926.64
31	8767.32
32	9752.31
33	592.86
34	14387.52
35	330.18
36	656.61
37	1761.72
38	4000.02
39	417.36
40	1317.63
41	9642.93
42	1003.08
43	1600.05
44	7669.71
45	6217.17
46	11341.92
47	3191.28
48	2498.88
49	2335.95
50	2055.63
51	9187.35
52	21294.63
53	7659.09
54	7798.44
55	8642.40
56	12632.67
57	10595.88
58	8325.42
59	11148.21
60	24321.96
61	5985.15
62	7226.94
63	1631.13
64	1800.78
65	3533.40
66	7681.80
67	253.35
68	6478.62
69	1206.63
70	4931.97
71	8270.40
72	5946.90
73	364.41
74	9062.43
75	1462.98
76	2071.59
77	19088.28
78	3946.53
79	2899.50
80	3985.74
81	3273.39
82	1510.02
83	3928.29
84	2146.98
85	11492.40
86	24045.78
87	1847.97
88	21855.09
89	2228.76
90	17952.60
91	21122.85
92	13.74
93	9.24
94	12.24
95	10414.62
96	18959.67
97	1980.90
98	6375.84
99	914.52
100	17453.97

Mean	5804.23
Standard Deviation	7290.06

1600m^2, 5 Robots

S/no.	Time
1	12926.58
2	8798.67
3	2406.81
4	1640.01
5	936.48
6	4020.15
7	4540.65
8	1122.30
9	1855.77
10	4904.88
11	4423.50
12	11943.00
13	5210.10
14	5669.10
15	1738.50
16	1517.28
17	17439.21
18	2369.49
19	174.57
20	2073.24
21	23929.32
22	4138.05
23	43.23
24	43.02
25	42.03
26	2939.19
27	1943.31
28	6084.42
29	3957.81
30	2274.12
31	8054.61
32	609.72
33	985.68
34	851.46
35	3637.11
36	15765.93
37	4628.73
38	1680.06
39	3821.79
40	4001.40
41	5112.48
42	11335.74
43	3508.89
44	7993.77
45	501.84
46	2062.26
47	17478.42
48	4556.85
49	12855.81
50	106.41
51	93.75
52	128.91
53	1217.82
54	3836.46
55	10864.29
56	841.32
57	6644.64
58	6186.15
59	16799.28
60	3898.14
61	4679.70
62	9951.24
63	270.69
64	10185.39
65	24512.76
66	3236.88
67	1598.22
68	3715.95
69	7533.15
70	3041.46
71	4453.35
72	673.47
73	849.03
74	5666.28
75	4101.69
76	4191.54
77	5790.33
78	24731.79
79	21.21
80	13.71
81	15.78
82	29.55
83	19.50
84	21.21
85	8299.95
86	5291.16
87	14236.98
88	2120.64
89	7240.47
90	4840.68
91	14358.72
92	524.73
93	9466.89
94	1749.69
95	2371.20
96	7627.14
97	752.55

Mean	5147.93
Standard Deviation	5533.00

1600m^2, 6 Robots

S/no.	Time
1	4807.20
2	822.18
3	1198.98
4	14223.72
5	2182.65
6	9272.40
7	14949.18
8	6111.93
9	14235.00
10	7527.24
11	4542.24
12	1519.71
13	10083.51
14	17562.03
15	10479.66
16	13702.47
17	19686.78
18	26011.95
19	11719.11
20	14784.21
21	5993.55
22	6034.20
23	1213.95
24	22105.68
25	5332.44
26	5251.44
27	2343.81
28	12505.29
29	807.90
30	6932.58
31	1860.06
32	3317.85
33	9618.81
34	6349.11
35	625.44
36	5175.33
37	6288.57
38	2883.36
39	3777.33
40	3219.39
41	2171.73
42	1564.80
43	17846.10
44	14918.91
45	2842.95
46	4917.00
47	7092.18
48	605.07
49	11469.63
50	8294.31
51	1273.53
52	5221.92
53	8737.02
54	8594.70
55	5210.43
56	5749.89
57	4678.44
58	3009.87
59	6208.11
60	8299.95
61	16815.42
62	18023.46
63	14950.17
64	12405.48
65	1234.74
66	13425.90
67	7556.67
68	3273.57
69	6933.33
70	3860.49
71	8356.32

Mean	7783.10
Standard Deviation	5741.88