Adaptive Formation Control for Heterogeneous Robots With Limited Information

by

Michael Andrew Rolland de Denus

A thesis submitted to
The Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements
of the degree of

Master of Science

Department of Computer Science
The University of Manitoba
Winnipeg, Manitoba, Canada
April 2013

Thesis advisor | Author

John E. Anderson | **Michael Andrew Rolland de Denus**

Adaptive Formation Control for Heterogeneous Robots With Limited Information

Abstract

In many robotics tasks, it is advantageous for robots to assemble into formations. In many of these applications, it is useful for the robots to have differing capabilities (i.e., be heterogeneous). These differences are task specific, but the most obvious differences lie in sensing and locomotion capabilities. Groups of robots may also have only imperfect or partially-known information about one another as well. One key piece of information that robots lack is how many other robots are in the environment. This thesis describes a method for formation control that allows heterogeneous robots with limited information to dynamically assemble into formations, merge smaller formations together, and correct errors that may arise in the formation. The approach is shown to be scalable and robust against robot failure, and is evaluated in multiple simulated environments.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to begin by thanking my advisor, Dr. John Anderson, meeting with you weekly to discuss my progress has helped me more than you know. I would also like to thank my committee, who will be reviewing this thesis. In addition, I would like to thank my parents, for pushing me to complete my work.

Finally I would like to thank my girlfriend Cara, for putting up with the long hours as I worked full-time, and spent many evenings on my thesis.

This thesis is dedicated to Cara, Your love and support have helped me through this difficult task.

Chapter 1

Introduction

1.1 Introduction

This chapter is an overview of the motivations of my thesis, an introduction to the problems that my research addresses, and an overview of my methods for addressing them. It outlines some of the problems with current approaches to formation control, and explains how my approach deals with them. This chapter also presents my research questions, which will be answered in section 5.3.

1.2 Motivation

In many situations where individuals must travel or perform tasks in groups, it is advantageous to have these individuals form and maintain physical formations. Such formations have many benefits over uncoordinated individual movement. These benefits are situational, and depend on the formation chosen. Ideally, robots will choose a formation that complements their goals. One benefit is protection from outside attack. The individuals closer to the centre of some formations, such as squares, are far less vulnerable than those on the perimeter. This can be seen in several real-life situations, such as soldiers marching in formation. Similar examples are a herd of cattle or a flock of birds. These are far less precise than the square formation of soldiers, but they serve a similar defensive purpose Goldman [1980]. Another benefit of using formations is the natural division of search tasks. When searching an area, it is more effective to send a parallel line of people through the area than to send people off randomly. This formation not only ensures coverage of the area, but reduces redundant search Balch and Arkin [1998]. Yet another benefit is the increased certainty of the positions of other nearby individuals. This reduces the complexity of planning a path. One final benefit is that robots will not likely be in the desired path of other robots. This is known as a reduction in interference Mataric [1992a]; Rosenfeld et al. [2008], as robots are less likely to need to modify their paths to avoid others once in formation. Given the real-world utility of formations, it is important to have a formation control technique that can be used effectively in real-world conditions.

Robot *heterogeneity* is the use of multiple robots with differing capabilities, or physical properties. In many real-world problems, regardless of whether formations are employed, heterogeneity is useful. For example in a search domain, it may be useful to have a robot capable of retrieving the object, but it would not necessarily make sense to give all robots such a capability. Two reasons that often arise are cost Howard et al. [2006] and specialization Yamauchi [2004]; Kiener and von Stryk

[2007]. When building large teams of robots, making some of them less expensive can be a large motivator Howard et al. [2006]. Using combinations of less expensive sensors can produce positive results, as shown by Burion [2004]; Howard et al. [2006]. Specialization can also be a motivator when dealing with widely varying tasks Yamauchi [2004]; Kiener and von Stryk [2007]. Either of these types of heterogeneity may be applicable under my approach. One example is using a formation to protect a specialized robot as it moves to a destination. Another is to use a formation containing a small number of robots with very powerful sensors, and a larger number of robots with weaker sensing to quickly map an environment, as in Howard et al. [2006].

It is infeasible in many circumstances of a robot to have complete knowledge of all other robots in the environment. This would require communication with infinite range, and would greatly increase the number of messages that need to be handled, as described by Yoshida et al. [1994b]. It is also trivial to see why perfect knowledge of the environment is impractical. Sensors have limited range, and do not always return perfect data. Some of these limitations are described in Burion [2004]. All of these factors combine to limit the amount of information that a robot could possibly have regarding its environment, and other robots. An approach that works with this limited level of information is far more likely to be successful in real-life situations than one that assumes full knowledge.

One key element that is important in an approach to formation control is the ability to have robots join and leave the formation. There are many reasons why a robot may leave a formation. It may be separated from the formation by obstacles, it may break down, or be destroyed, etc. New robots may be introduced to the environment at any time. The operators may decide that more are needed. A robot may rediscover the formation after becoming lost, etc. Most approaches do not touch on this topic directly, though a few handle it implicitly, by not making reference to specific neighbours Lee and Chong [2007]; Dieudonné et al. [2008], and not using central processing. Once an approach allows for the addition and removal of robots, several new problems are introduced. The first such problem is the need to maintain the proportions of the formation. If a robot is lost from one location in the formation, how do the others need to re-arrange to fill its place? Similarly, how do robots react to the addition of a new formation member?

When handling dynamically constructed formations, there are several additional problems that arise. The first problem arises when robots all join at one place in the formation. This can result in a formation that looks nothing like what is intended. For example, a formation intended to resemble a V can instead resemble a line, if all robots join on one side. This need for re-arranging, or balancing is a major complicating factor in the creation of formations. Some techniques Lee and Chong [2007]; Dieudonné et al. [2008] are able to handle this elegantly, by maintaining simple rules which implicitly do this balancing. This sort of balancing in more complex formation is a problem that my technique addresses.

Yet another problem that arises when handling dynamic formations is the requirement to merge smaller formations into one large one. There are several challenges, including selecting which formation is in control of the merging procedure, and choosing which robots go where. This can be a very complex process, involving a large re-arrangement of robot relationships. This is difficult, because robots need some way to synchronize. Without global communication, this becomes even more complex. Robots potentially have incomplete information on their own formations.

Finally, in order to ensure the development of a single formation, it is useful for robots to have some way to search for distant neighbours. This could potentially allow smaller formations that are not in visual range of one another to merge into a single larger formation.

A large amount of research has been done in the area of formation control. I will discuss many examples in Chapter 2. Many of the weaknesses in current formation systems arise because the creation of a formation affects all member robots. There are many tradeoffs to be had between central processing, global communication, and robot-level synchronization. The existing solutions tend to have one or more of the following weaknesses:

1. Lack of robustness against failure Fredslund and Mataric [2002]; Gilbert et al. [2009]; Consolini et al. [2009]; Ray et al. [2009]; Dierks and Jagannathan [2009]; Brandao et al. [2009]: If one robot in the formation fails, it can disable a large portion of the formation with it. In some systems, the failure of a single lead robot could disable the entire formation. This lack of robustness puts these systems at a great disadvantage in the environments where formation movement is most often used.

2. Lack of scalability Hattenberger et al. [2007]; Brandao et al. [2009]; Rampinelli et al. [2009]; He and Han [2009]; Kitts and Mas [2009]: Systems that employ central processing or global communication tend to suffer from a lack of scal-

ability. Those systems that rely on central processing obviously depend on the processing power and network bandwidth of the central processor. Global communication has been shown to be impractical beyond a small number of robots Yoshida et al. [1994a]. This is due to the exponential growth in the number of messages as the number of robots increases.

3. Global knowledge requirements Balch and Arkin [1998]; Fredslund and Mataric [2002]; Gilbert et al. [2009]: Many systems require some shared global information. These techniques vary greatly. One example is the assignment of fixed neighbours. This greatly reduces the flexibility of the formation, and makes it less robust against failure. Another example is knowledge of the positions of all others in some sort of shared map. This makes the computation of desired positions very simple, but is not at all realistic in the real world.

4. Unrealistic environments: Formation systems tend to be tested in very forgiving environments. These test environments have uniform terrain, no communication or sensor interference and few obstacles. These types of environments simply do not exist. For a system to be successful outside of the laboratory, it must be evaluated in real-world conditions, involving all of these factors.

5. Lack of adaptability Fredslund and Mataric [2002]; Brandao et al. [2009]; He and Han [2009]; Kitts and Mas [2009]; Rampinelli et al. [2009]: Many systems can't deal with the dynamic addition or removal of new robots, or sense and deal with robot failure. For real-world operation, it is important that a formation system be adaptable.

6. Simplicity of possible formations Lee and Chong [2007]; Dieudonné et al. [2008]; Moshtagh et al. [2009]; Kurabayashi et al. [2009]; Brandao et al. [2009]; de Denus et al. [2010]: Systems which suffer from none of the above weaknesses tend to be very limited in the type and complexity of formations that they can support. For example, those based-on simple geometric rules tend to only support one fixed formation.

1.3 Overview

My work focuses on robot heterogeneity within the formation, while emphasizing a decentralized approach. Heterogeneity complicates the act of creating a formation and maintaining it: If it can't be assumed that all robots have the same level of ability (e.g. perception), the approach must change to accommodate the strengths and weaknesses of the individuals. My research provides an answer to the following question: Can the scalability, robustness and adaptability of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? I achieved this through the design and implementation of a new approach to formation control. This approach is designed to allow robots with greater abilities (better perception, more finely grained control) to compensate for the failings of weaker robots, with minimal use of global information.

My research addresses all of the weaknesses described in section 1.2, by sacrificing perfection. I employ a heuristic approach to formation control, employing simple local rules, local communication, and limited relaying of messages. In this approach,

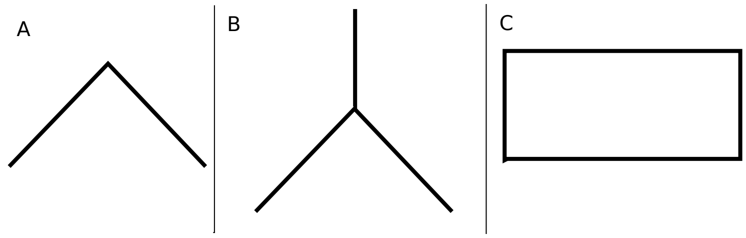


Figure 1.1: Panel A illustrated the V formation, panel B illustrates the formation with three segments meeting at one point. Panel C illustrates the rectangular box formation.

all robots are peers, and each has its own individual sensor data and internal map of the world. No single robot is essential to the continuation of the formation. This decentralized approach is designed to improve both the robustness and scalability of the system. There are two pieces of global knowledge in the system. The formation currently in use is the first piece. This small piece of global knowledge is necessary, since varying the selected formation at runtime is beyond the scope of my research. I did, however, prepare a list of formations to test the various components of the approach. These included a V formation, a rectangular box formation and a formation with three lines meeting at one central point. These formations are illustrated in figure 1.1. The second piece of global information is a shared destination. It is reasonable to believe that robots in close proximity would receive similar environmental cues, and want to move in similar directions. As I am testing the formation abilities of the robots, it is justifiable to study only the subset of robots that have a common goal. Any formation member not sharing this goal would not stay a member for long.

My approach limits the points of entry into a formation. The methods used to do so are described in section 3.2.1. The result is an approach that requires few robots to have knowledge of anything more than their immediate neighbours.

My approach supports the dynamic introduction of robots to (and removal of robots from) the formation. The system adapts quickly to the insertion or removal of a new robot. This in and of itself partially addresses the problem of robustness against failure. If no robot is essential, and leaving the system is possible, all that remains is the detection of failure. My approach also handles certain levels of partial failure. If a robot's wheels become stuck, and it can no longer move, it will eventually fall out of range of the formation. If a robot loses its ability to communicate, it will also eventually be dropped from the formation. In the current implementation, robots trust the messages that they receive. A malicious or malfunctioning robot could potentially cause harm to the formation, as long as it sent messages that were not obviously inconsistent.

In my approach, robot heterogeneity has two dimensions. The first dimension of heterogeneity is a robot's perception. Robots have varying visual ranges and varying ability to identify other robots. The other dimension of heterogeneity is a robot's locomotion ability. Some robots have strong motors, while others have weaker motors, or are carrying more weight. This is simulated by giving each robot a different maximum speed.

My approach does not assume perfect communication, and is able to recover from lost messages. It can tolerate temporary drops in communication, but does require that communication be restored within a few seconds.

1.4 Research Questions

To expand on the question posed in section 1.3, my question has three main components, each consisting of a separate research question.

The first question becomes: Can the scalability of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? In other words, how does heterogeneity affect the ability of robots to maintain formation, as the number of robots increases. A highly scalable approach should behave consistently with increasing numbers of robots.

The next question is: Can the robustness of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? This question addresses how robots deal with failures of other robots in the formation. A highly robust method tolerates robot failure with little to no impact on the formation.

The final question is: Can the adaptability of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? This question addresses the ability of robots to function in different environments. A highly adaptable approach will function well in environments with varying degrees of obstacle cover.

1.5 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 is the background chapter, discussing current and past research into similar topics, and comparing them to my approach. Chapter 3 is the Methods chapter, it discusses the details of my approach as well as the implementation that I chose. Chapter 4 is the evaluation chapter, it contains my experimental design, as well as qualitative and quantitative analysis of my results. Finally, Chapter 5 contains my contributions, and a concluding summary of my findings.

1.6 Summary

This chapter serves as an introduction to my approach to formation control. It outlines some situations when formations are important, and how robot heterogeneity is a natural part of many of these situations. Real-world problems that can be addressed through formation often demand some level of robot heterogeneity (e.g. the environment described in Harabor and Botea [2008]). The use of limited information is also introduced. This chapter describes some of the limitations of existing approaches to formation control. Finally, the chapter outlines my approach to formation control, and describes the specific aspects of formation that I examine in the body of the thesis.

Chapter 2

Background

2.1 Introduction

In a formation, robots maintain relative positions corresponding to some global pattern. They do this by maintaining a distance and angle from a reference. That reference can be a neighboring robot, a globally-selected centre, or some global leader Balch and Arkin [1998]. Most of the techniques summarized below use either a leader or neighbouring robot as a reference point. They all require varying amounts of global data and/or restrictive assumptions in order to operate. I will begin by reviewing techniques relying on central processing to calculate robot positions. Next, I will discuss leader-based techniques. Finally, I will address the diverse list of techniques based on local rules.

2.2 Definition of a Formation

A common way of defining formation is the use of a control law, which defines a spatial relationship between robots one example of such a definition is Moshtagh et al. [2009]. Another way of defining a formation is as a task assignment problem (e.g. Viguria and Howard [2009]), where each task represents maintaining a spatial relationship between 2 or more robots. Yet another definition involves the establishment of a common reference point, and defining spatial relationships with respect to this reference point as in Balch and Arkin [1998]. The common elements of all of these definitions are spatial relationships. As discussed in section 2.1, this relationships can be between multiple robots, or between robots and some point in space (e.g. Balch and Arkin [1998]). The important aspect of this definition is that a robot must maintain some spatial relationship with respect to this reference point, which may or may not be another robot. My approach involves defining groups of robots with similar spatial relationships, and grouping them together. I describe this approach in detail in section 3.2.1

2.3 Central Processing

Central Processing involves doing some amount of the computation for the entire formation in a single place, whether that be on a single robot (e.g. Hattenberger et al. [2007]) or by some external controller (e.g. Rampinelli et al. [2009]) The techniques in this section have varying degrees of reliance on some sort of central processing. This limits scalability, and often introduces single points of failure.

The work of Hattenberger et al. [2007] has been shown to be adaptive and performs well in simulated environments. The authors admit, however, that due to the central processing that takes place on one robot, the technique is not highly scalable. My approach achieved similar results while distributing the processing load.

He and Han [2009] developed a formation controller which is somewhat decentralized, but still requires the use of a single master controller. The cluster-space method employed by Kitts and Mas [2009] has a similar partially-decentralized design. These two approaches give each robot some degree of independent control, but are ultimately controlled by a single master. My system has no master controller, and is able to maintain formation independent of the number of robots.

Rampinelli et al. [2009] propose a system based primarily on a path calculated by a central controller. This is not feasible from a multi-agent systems approach, but is still interesting in its support for variation in the types of robots used. The authors address the problem of heterogeneity by adding control parameters that can vary based on the type of robot in use. I integrated a similar technique into my work, while remaining decentralized. I also focus on heterogeneity in sensing, which their technique does not address.

2.4 Leader-Based Approaches

Central processing is generally not very scalable or robust. One alternative is the use of one or more lead robots, and a series of follower robots. This immediately simplifies the problem of establishing a coordinate system for specifying desired robot positions, as all coordinates can be calculated relative to the leader. Various levels of autonomy are given to the followers. The works which follow take this general approach.

Kwon and Chwa [2012] created a formation control method that divides robots into subgroups. Each of these subgroups then has a defined relationship to other subgroups. This hierarchical method allows for groups of robots to be represented as a single lead robot. Robots within the subgroup are controlled by the subgroup leader, and travel beside the leader, at a specified distance. A formation consists of collections of these subgroups, with each subgroup leader following the leader of the subgroup in front of it. This approach is somewhat tolerant of missing robots, but they would cause local distortions in the shape of the formation.

Consolini et al. [2009] propose a formation control method with multiple levels of leaders, arranged in a hierarchical fashion. The relevant concept in their work is the fact that multiple robots assume some level of leadership in the formation. Their approach suffers from one of the primary failings that most leader-based formations have: It lacks dynamic selection of the leader. My approach addresses this limitation.

Ray et al. [2009] implemented a formation control system involving a master robot, used to define the front of the formation. The robots making up this front then are used to guide columns of other robots through obstacle-filled environments. Broadcast communication from a global lead robot is used to identify the column lead robots. This introduces a single point of failure into the system. Another limitation is that this system is only capable of producing formations consisting of a series of columns. My approach has neither of these weaknesses.

Dierks and Jagannathan [2009] created a leader-based system, consisting of a central leader and two chains of followers. One reason for the simplicity of the system is to allow robots to plan paths that avoid obstacles while maintaining formation. This would be a desirable feature for my system to have. It would likely require some sort of pre-planned path, but is not completely infeasible under my approach My system is also not be leader-based, and thus does not have this point of failure.

Shin et al. [2007] use a traditional leader-based formation technique, but study the effects of unreliable communication and methods for dealing with it. My system has some reliance on inter-robot communication, and communication failure is possible. For this reason, I employ a similar method of falling back on sensor-based navigation when communication fails. My approach however, requires that communication be restored at some point.

2.5 Global Information

Some formation control techniques rely on some body of prior knowledge. This could be a map of the environment, knowledge of the identities of others, knowledge of the absolute positions of others, etc. This added knowledge increases performance. It generally does so at a cost, by either greatly limiting the applicability of a system to the real world, or introducing points of failure into the system.

Some early formation control systems involve global knowledge (e.g. Balch and Arkin [1998]). These systems, while they enjoy good performance and are a good starting point for research in formation control, would not be possible to implement in a realistic environment. Other systems rely on chains of intercommunicating robots (e.g. Fredslund and Mataric [2002]). These techniques still have the requirement that each robot have a predetermined neighbour, and occupy a specific place in the formation. This is a limitation that my system does not have. I use a similar chain-based technique, but the neighbours are determined dynamically and may change over time.

Gilbert et al. [2009] describe a formation control technique which is robust against unforeseen changes in the number of members. They do so by breaking up responsibility for maintaining the formation across multiple coordinating robots. Each of these robots is responsible for the proper maintenance of the formation within a given geographic area. This essentially creates a global coordinate system, making the establishment of formations much simpler. These coordinating robots are also a possible source of failure. If one fails, it can disable an entire region of the formation. It is however, more robust than the use of a single leader. My technique uses a different type of coordinating robot, but all robots in the system are interchangeable, making my approach more robust against failure.

2.6 Local Rules

Another approach to formation control is to attempt to establish a set of rules governing the formation. These rules can be established beforehand, or determined dynamically. Once the rules have been established, robots use their sensory information to attempt to satisfy these rules. In general, these rules tend to be fairly simple to compute. A wide variety of formation control techniques make use of these sorts of rules.

Lee and Chong [2007] describe a method of establishing a formation of robots consisting of a series of triangles, while Yamaguchi [1997] developed a technique for creating a curve to block a doorway, or other similar structure. These types of formations are excellent for a specific purpose, but the techniques cannot be applied to produce a larger variety of formations. They do, however give valuable ideas for maintaining formations using only local data.

Dieudonné et al. [2008] present a technique for establishing a formation consisting of a pair of concentric circles. As with my intended approach, theirs does not require global communication. In addition, they do not make use of any global information and their technique is limited to simple fixed formations. My technique maintains these strengths without limiting the number of possible formations in the same way.

Brandao et al. [2009] proposed a method involving control of a 3-robot formation. This technique relies on tracking and compensating for variations in distance and angle between robots. It is able to successfully lead a formation along a pre-defined path, while varying the formation shape. The main limitation of this approach is that the control algorithm is strongly tied to the fact that there are only three robots.

The approach of Moshtagh et al. [2009] is very interesting. Robots rely purely on visual feedback, making this technique scalable and robust. The number of different formations, however, is limited. The control law used in this approach limits the number of possible formations that can be formed. Only circular and linear formations are possible with this approach. My approach maintains the decentralized nature of this control strategy while increasing the number of possible formations.

Kurabayashi et al. [2009] use a technique based on measuring the variations in oscillators to determine headings and positions of neighbours. This method requires no identifiers for individual robots, and positions are determined dynamically at runtime. This approach is limited in the number of possible formations that it can support.

Hadaegh et al. [2001] describe a technique that allows the dynamic removal and re-addition of formation members from a fixed formation. The technique uses a rule-based system, and observations of all other robots to determine course corrections necessary to maintain formation.

My previous work de Denus et al. [2010] employed an approach to formation control where rules for distance and angle from a neighbour are selected probabilistically from a list of choices given a neighbour's state. This method does not consider heterogeneity beyond the ability to perceive one's neighbour. It also biases robots towards joining the longest regions of the formation, which hinders the development of other parts of the formation. This prior work was designed entirely by me, with advice on choices provided by my co-authors. My new approach introduces two new concepts designed to combat the latter problem. First, joining is restricted to certain locations. Secondly, the relative length relationships between different parts of the formation are monitored, and the formation is adjusted to compensate for imbalance. My approach, described in section 3.2, also makes use of a much greater degree of heterogeneity in robots, and has been evaluated in more complex environments.

2.7 Robot Heterogeneity

Howard et al. [2006] have developed an approach to mapping, which uses simple leader-based formations to deploy a sensor network. Each leader robot is outfitted with powerful and expensive sensing equipment, while the followers have much simpler sensing equipment. The robots in their trials were able to complete their goals, and maintain a basic formation, despite the limitations of the simpler robots. My approach also explores the reduction in sensing capacity of a subset of robots.

An example of this is presented by Yamauchi [2004]. In this research, the same physical platform, the PackBot, is outfitted with a variety of sensors, manipulators and locomotion aids. This makes it suitable for several different tasks, including flight, object retrieval, and specialized sensing. Sensing specializations are also discussed by Burion [2004]. He demonstrates that combinations of multiple sensors can be effective in identifying human bodies. Cost is considered, and it is noted that outfitting a single robot with all sensors would be impractical from a cost standpoint. Although my approach does not specifically deal with specialized sensors, I do vary the sensing abilities of my robots. This could simulate specialized sensing, which is well-suited or poorly-suited to navigation.

Harabor and Botea [2008] address two more important elements of robot heterogeneity. These elements are size, and ability to cross terrain. They achieve positive results in planning paths for robots of differing sizes, and abilities to cross terrain. The ability to cross difficult terrain can be approximated under my approach by varying the locomotion abilities of the robots. Robot size, on the other hand, is not something that my approach addresses directly.

Robot heterogeneity is not necessarily physical. In some purely simulated approaches, such as Dutta and Sen [2003] we see heterogeneous robots, with differing levels of ability to perform a task. This could be seen as a representation of physical qualities, but does not map directly to physical properties. These sorts of high level differences are sometimes possible to approximate using my approach, depending on how well they map to actual physical capabilities.

Under certain approaches, such as Kumar and Garg [2011], robot heterogeneity is not due to the physical properties of the robots. It is instead due to different labels, applied to otherwise identical robots. My approach makes an attempt to keep track of the differences between a robot and other nearby robots, but it does so by keeping track of their physical properties.

My approach to formation control allows for heterogeneity in sensing, and heterogeneity in locomotion ability. This allows my approach to approximate most of the types of robot heterogeneity described in this section.

2.8 Behaviour-based Robotics

A behaviour-based approach is well suited to my domain, because of the many different goals that robots in a formation need to balance. Robots need to maintain formation, avoid obstacles, move towards some goal, communicate with others, etc. A behaviour-based approach Arkin [1998] allows the definition of behaviours, which map from perception to action. Mataric [1992b] explains how this technique is more generally useful than a purely reactive approach. It allows for interaction between behaviours, and allows for some level of internal state to be maintained. Arkin [1998] describes a general framework for behaviours that is used very commonly in behaviour-based approaches. He breaks behaviours down into two classes: Perceptual schemas, and motor schemas. Perceptual schemas take in information for a robot's sensors, and process it, producing an intermediate result, which can be used as an input to a motor schema, or to another perceptual schema. The motor schema aggregates the results of its perceptual schema inputs, and produces an output designed to be delivered to a motor. The outputs of all motor schemas are aggregated, and sent to the motors that control the robot Arkin [1998]. My approach follows this pattern, having a single motor schema, which controls the wheels of the robot, and multiple perceptual schemas, which provide heading information. This is a simple and effective way of managing the many goals of a robot in formation. Another successful behaviour-based approach is Balch and Arkin [1998].

2.9 Communication

There are many properties of robot communication that can lead to differences in how robots communicate. One important distinction is whether communication is global or local. Yoshida et al. [1994b] discuss this distinction, noting that global communication does not scale well to large groups. My approach makes use of local communication, meaning that each robot sends messages to a single target, as described in Yoshida et al. [1994b].

Another important facet of communication is the communications medium. Some communication is visual, and requires that a robot observe the sending robot to receive it (e.g. Wang [1993]). Communication can also occur through changes to the physical environment (e.g. D'Angelo and Pagello [2005]). Communication can also use some sort of communications network, as described in Legras and Tessier [2004]. In their approach, there is the possibility for other robots, who are not the intended recipients, to read the messages. My approach makes use of networked communication, but does not support the interception of messages intended for other robots.

When communicating information form a local perspective, it is important that the information be translated into some form that will make sense to the receiver. This concept is known as grounding Billard and Dautenhahn [1999]. When communicating information about the environment or positions of other robots, it is important to provide some sort of grounding. Other approaches use meeting places e.g. Wiebe and Anderson [2009], or teaching of pre-labeled locations e.g. Billard and Dautenhahn [1999]. My approach avoids this issue by not communicating positions of objects between robots. It instead transmits more general messages indicating whether or not a certain object is visible. This helps to avoid the grounding problem, while still giving a small amount of relative positioning information.

2.10 Summary

In this section, I have given an overview of current approaches to formation control, and the handling of heterogeneous robots. I have addressed how each of these formation control approaches applies to my research questions. I have also given some background on heterogeneous robots, and behaviour-based robotics, as both of these topics feature in my approach to formation control.

Chapter 3

Methods

3.1 Introduction

As noted in chapter 1, I am attempting to demonstrate that robots can dynamically assemble into formations, with no prior knowledge of the number or types of other robots in their immediate vicinity, no common coordinate set, no central control, and no single robot as a point of failure. I intend to show that this is not only possible, but that it is also scalable, robust against failure and adaptive to changing environmental conditions. To this end, I have implemented an approach to formation control that exhibits all of these qualities. This approach is described in the remainder of the chapter.

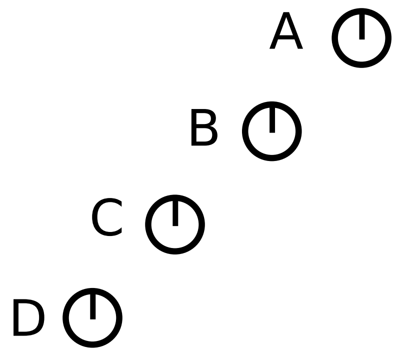


Figure 3.1: An example of a segment. Robot B has two neighbours, they are robots A and C. Robot B's target is robot A.

3.2 Approach

3.2.1 Preliminaries

My approach to formation control assumes that all robots have the desire to participate in a formation. It also assumes that participants possess a mechanism for agreeing upon a formation, and sharing it with new members. This is not an unreasonable assumption in the cases discussed in Section 1.2, where formations are useful.

In my approach to formation control, I define a *formation* as an assembly of *segments*, each of which consists of a series of robots. A sample segment is shown in Figure 3.1. In order to understand these elements, a few key concepts are required. First, the *base length* of the formation is the measure by which all other lengths in the formation are expressed. For the sake of simplicity, I define base length as being equal to the relative length of the shortest segment. Once the base length has been chosen, all relative lengths are expressed as multiples of the base length. The fact that formations use relative lengths means that there can have an infinite number of

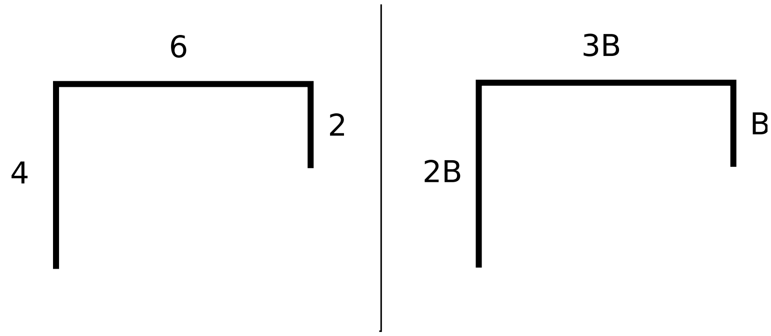


Figure 3.2: The left panel shows the raw relative lengths of segments in the formation. The shortest segment has length 2, its length is used as the base length. All lengths are expressed as multiples of the base length, denoted by B.

representations of the same formation. Expressing all lengths as multiples of a base length gives a common representation, which is more useful when comparing multiple formation specifications. In calculations involving relative lengths of segments, the base length can be treated as a length of 1. The concept of base length is illustrated in Figure 3.2. Second, two robots are said to be *neighbours* if one is using the other as a reference. Third, a *neighbour set* is the set of segments that intersect at a point. The robot at this point of intersection is considered a member of all of the intersecting segments. The neighbour set can be thought of as the set of all segments to which that robot belongs. In this approach, it is valid for a robot to be a member of multiple segments.

Segments are the basic building block of more complex formations. Members of a segment maintain a relative distance and angle from their neighbours within that segment. This distance and angle are each controlled by a function. The angle produced by the angle function is always relative to the centre of the robot's field of view. Because of this, some sort of external reference point or goal is necessary to guide the formation members. This is simulated by the goal seeking behaviour, which I will discuss in section 3.3.1. If the formation is not in motion, the angle function is not an effective way of ensuring proper formation, since there are many positions around a robot where the combination of distance and angle functions can be satisfied. Motion of the formation ensures that the robot's local view of the formation is consistent with the global view. The distance and angle functions take the current distance and angle as inputs, allowing for flexibility when determining how robots position themselves relatively. These functions could be used to create curves, or even vary the shape of the segment over time. Segments have a *length* field. This field expresses the desired relative length of the segment as a multiple of the base length. These length values are used solely to construct length ratios between segments, and have no real world units. I will discuss later how these ratios are used to balance the formation, without necessarily knowing the value of the base length.

I define an *entry point* as the endpoint of a segment, and an *entry point robot* as a robot occupying an entry point position. The role of each entry point robot is to control access of new members to a given segment, and by extension, the formation. These are the robots that occupy the positions where multiple segments converge to form a neighbour set. Each entry point robot maintains approximate counts of the number of robots in each neighbouring segment, in order to perform balancing according to the length ratios calculated from the formation definition. They maintain these estimates through a combination of communication and perception. The methods I developed to maintain counts are discussed in Section 3.3. When a robot requests entry into the formation, the approximate counts are used to determine the action to be taken. Any robot becomes an entry point robot if it has fewer than two neighbours, or if it has multiple neighbors that do not share segments. The robot ceases to be an entry point robot if either of these cases ceases to be true.

For each entry point I define a *neighbour set* to be the set of all segments that terminate at this entry point. Each segment has two such sets, one for each entry point. I call these sets the *leading neighbour set* and *trailing neighbour set*. The distance and angle functions combine to create a natural direction to each segment, making it possible to define one entry point as leading and one as trailing. When establishing a new formation, the entry point robots are responsible for determining how new robots are to be allocated to new segments. They use the neighbour set information to determine the segments that they are eligible to create. Each entry point is considered to be a member of all segments in its associated neighbour set.

To support communication between robots, I define an inter-robot communication protocol. Each message begins with the message type, the ID of the sender, and the ID of the receiver. Upon receiving a message, a robot can either forward it on, or reply to it. The action taken depends on the message type. In my approach, communication is never assumed to be perfect: Application circumstances may make communication unreliable, and communication should be useful to the degree it is available.

To support robot heterogeneity, robots need to be able to assess their own abilities in certain areas as well as the abilities of others. The abilities of all other robots are initially unknown. Robots directly communicate with other robots to make up for some of their lack of sensing ability. Robots can transmit details of their locomotion

Segment ID	Leading Neighbour Set	Trailing Neighbour Set	Relative Length	Distance Function	Angle Function
0	1	3	1	d=1	a=0
1	0	2	2	d=1	a=$\frac{\pi}{2}$
2	1	3	1	d=1	a=0
3	0	2	2	d=1	a=$\frac{\pi}{2}$

Table 3.1: The formation definition for a rectangular box formation

capabilities, as far as unreliable communication allows. Robots store this information, and can use it to slow down, or speed up their motion. Robots may also transmit requests for sensing data. Because of the potential differences in coordinate sets, exact object positions are not relayed. Instead, the receiving robot returns whether or not the object in question is visible.

The representation that I use for formations allows a formation to be specified as a collection of interconnected line segments. Segments can be made up of straight lines or curves. The only notable limitations are that the number of segments must be finite, and that each robot can have only one target. For example, Table 3.1 shows a formation definition for a simple rectangular box. The box is the outline of a rectangle, twice as wide as it is long.

There are five possible actions that a robot can take with respect to the formation. These actions are joining, leaving, size estimation, balancing, and inconsistency detection. Joining is actively performed by a robot. Leaving the formation is implicit, and occurs when a robot fails or is separated from the formation. Balancing is an action performed by entry point robots only, and is intended to keep the various segments to which an entry point robot belongs at the proper relative lengths. These actions will be described in depth in the subsections that follow.

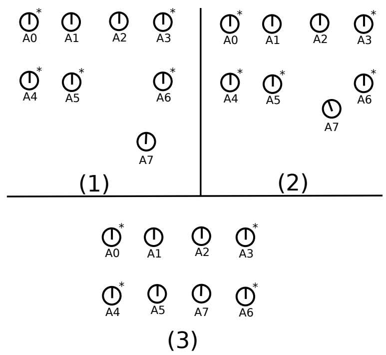


Figure 3.3: The joining process illustrated. The formation pictured is the rectangular box formation defined in Table 3.1. Entry points are marked with '*' (1) Robot A7 detects a nearby formation, initiating communication with A6. (2) A6, being an entry point, directs A7 towards its left side. (3) A7 Joins the formation. A5 recognizes that it is now part of a longer segment and drops entry point status.

3.2.2 Joining a formation

The three ways in which a robot may join a formation can be described as follows. In the first case, one robot encounters another, neither is participating in a formation. In this case, the two robots form the entry points of the longest segment of the formation. The first robot to request joining becomes the trailing entry point. The receiving robot becomes the leading entry point. This is arbitrary, but tends to result in less movement to achieve the desired distance and angle, as the requesting robot is more likely to be physically behind the receiving robot. If communication fails, robots continue to send messages to other visible robots, until communication is successful.

A robot may also encounter and join an existing formation. In the case of a robot encountering a non-entry point in an established formation, the entering robot sends a request for entry. This request will be denied, since joining at such a location is not permitted. As a part of the denial message, the responding robot suggests its own target as a possible place to attempt joining next. This process continues until the joining robot is accepted or rejected by an entry point robot. If the encountered robot is an entry point, the entry proceeds as follows: Let N_a be the number of robots in all segments in the neighbour set of entry point robot A. Let L_S be the relative length of segment S. Let L_{SA} be the sum of relative lengths of all segments in the neighbour set of entry point robot A. For each segment S in the neighbour set of entry point A, we calculate $P_s = \frac{L_s}{L_s a}$. P_s. This represents the percentage of N_a that belong in segment S. Multiplying P_s by N_a gives us the desired number of robots in segment S. After calculating this desired number of robots for each adjacent segment, the entry point robot identifies a destination segment. The destination segment is the furthest below its desired number of robots. The case of a robot joining the formation by communicating with an entry point is illustrated in Figure 3.3. In either of these cases, if no response is received, the joining robot follows the formation, and periodically attempts to join.

Finally, I address the case of the merging of established formations. This occurs when an entry point robot becomes aware of a member of another distinct formation. Methods for recognizing members of other formations are described in Section 3.3. Once such a robot has been located, the first step is determining which formation joins the other. The entry point robot requesting the joining attempts to contact an entry point robot in the other formation. If successful, they negotiate the joining. During negotiation, each entry point robot estimates the size of its formation as a whole, using the method described in subsection 3.2.4. The smaller formation then joins the larger one. To begin the merging, the entry point robot negotiating on behalf of the joining formation sends out a message to all of its neighbours. The message indicates a target robot in the other formation. As robots receive the message, they drop out of formation, and begin following their nearest neighbour, creating a line. At this point, the robots in this line are not considered to be in formation. Nearest neighbours are periodically re-evaluated, allowing more complex formations to break apart correctly. This line then moves towards the target robot, each member joining as an individual when it reaches the target robot. An robot not receiving this message simply continues on, possibly merging later. In the case of a miscommunication, or initiation of merging from multiple sources, both formations collapse, and reform into at most N formations, where N is the number of simultaneously initiated merging operations. My implementation involves some safeguards against multiple simultaneous merges, they are discussed in section 3.3

When joining or merging, it is possible that a message gets lost. In such a case, the formation may be left in an inconsistent state. In such cases, inconsistency detection, as described in Section 3.2.6 will attempt to resolve the issue.

3.2.3 Leaving a Formation

Leaving the formation is a special action, as it is involuntary. Robots leave only when failure or unsuccessful navigation (e.g. obstacles) forces them to. The robot's neighbours need to recognize its absence and attempt to repair the formation. To avoid some of the inconsistencies that can occur when attempting to blindly repair a formation, Any robot using this robot as a reference point splits off from the formation, forming a loose line formation with its nearest neighbor, this progresses down the chain of robots, possibly causing a large percentage of the formation to break apart. These robots then perform the merging operation mentioned in subsection 3.2.2. Over time, the error introduced by the robot's departure is corrected by the counting mechanism described in Section 3.3. If the robot in question is an entry point robot, its departure causes the robot that it was using as a reference point to become an entry point robot.

3.2.4 Size Estimation

Each entry point robot has a responsibility to maintain counts of all adjacent segments. These counts allow a robot to estimate the total formation size. The size, in this case refers to the number of robots in the formation. This is an estimate only, since we do not have the guarantee that each segment has exactly the correct number of robots at all times. The estimating robot first looks at the relative lengths and counts of all adjacent segments. It uses the following technique to determine the expected size of the formation, given the observed size of the formation. Let L_r be the sum of the relative lengths of all segments in the formation. Let A_{ca} be the total number of robots occupying each segment adjacent to robot A. Let A_c be the total number of robots in the formation. This sets up the relationship:

$$\frac{L_{ra}}{L_r} = \frac{A_{ca}}{A_c}$$

To find Ac, we solve, giving $\frac{L_r}{L_{ra}} * A_{ca} = A_c$ This gives us the correct counts, assuming all segments were separate, and contained no overlapping members. Because we consider entry point robots to belong to multiple segments, we need to subtract the number of robots that are counted as part of multiple segments. This occurs at any entry point. These entry points are easily determined from the formation specification. For each entry point, we must subtract S - 1 from the estimate total, where S is the total number of segments meeting at that entry point. let E be the set of all entry points, and S_{Ei} be the number of segments meeting at entry point i. The size estimate becomes:

$$A_c = \frac{L_r}{L_{ra}} * A_{ca} - \sum_{i=1}^{|E|} S_{Ei} - 1$$

3.2.5 Balancing

Balancing is the operation by which a formation moves robots from one segment to another. This can become necessary over time as robots join and leave the formation. As noted in subsection 3.2.1, the formation describes the relative lengths of each segment. At regular intervals, entry point robots examine the counts of all adjacent segments. For each adjacent segment, they calculate the difference between actual and desired segment length. These are calculated by summing up the number of adjacent robots, call this sum N_a. Next, the entry point examines the relative lengths of each adjacent segment. Let the relative length of segment S be L_S. Let the sum of relative lengths of all segments adjacent to entry point A be L_{SA} for each segment S adjacent to entry point A, we calculate $P_s = \frac{L_s}{L_s a}$. P_s represents the percentage of N_a that belong in segment S. Multiplying P_s by N_a gives us the desired number of robots in segment S. After calculating this desired number of robots for each adjacent segment, the entry point robot identifies a source segment and a destination segment. The source segment is the segment that is furthest above its desired number of robots. The destination segment is the furthest below its desired number of robots. In case

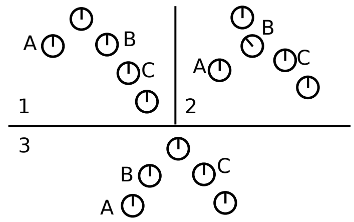


Figure 3.4: The three panels illustrate steps in the balancing process for a simple V formation, with two segments of equal length. Panel 1 shows the initial imbalanced formation, Panel 2 shows robot B changing segments, and the other robots adapting to this. Panel 3 shows the final balanced formation.

of a tie, one segment is chosen at random. Once the source and destination have been selected, the entry point calculates the imbalance. Imbalance is the minimum difference between desired and actual robot counts for the source and destination. If the imbalance is not at least one robot, we do not proceed with balancing, as doing so would create oscillation. Figure 3.4 illustrates one case of the balancing process. It shows how robot B changes segments, and how robots A and C adjust their targets. Note that balancing is a gradual process. Each balancing operation moves one robot between segments, but the balancing operation is repeated at regular intervals. Each iteration of balancing brings the formation closer to its ideal state.

The movement of a robot from one segment to another requires multiple messages to different robots. If at least one but not all of the messages are lost, inconsistency detection, as described in section 3.2.6 is necessary to resolve the problems that arise.

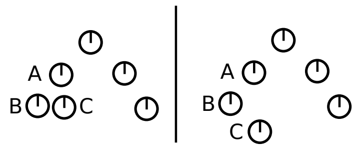


Figure 3.5: Two possible inconsistencies in a simple V formation with 2 segments of equal length. The left panel shows two robots, with the same target and segment. The right panel shows a robot that has joined in the wrong place.

3.2.6 Inconsistency Detection

During the initial establishment of the formation, and during balancing or joining, it is possible for several types of errors to arise. These errors typically arise in the case of a lost message. The first type of error is a single robot, which has two neighbors attempting to occupy the same segment. This condition is easy to detect, as in any such case, both of these robots must be neighbors of the robot in question. Since is is possible for robots to have as many as 2 neighbors occupying the same segment, we must make use of the fact that in all valid cases, one of these two robots must be the reference point of the robot. To detect this error condition, we iterate over each robot's neighbors, and count the number belonging to each segment. Next, we subtract one from the segment occupied by the robot's reference point. Any segment occupied by more than one robot is treated as an error. Neighbors from this segment are discarded until only one remains. The robots selected for removal are random. This can require that large numbers of robots rejoin the formation. This situation is illustrated in the left panel of Figure 3.5. Robots B and C are both following robot A, and are attempting to occupy the same segment. As a result, one of them will be instructed to leave the formation and rejoin.

The next type of error occurs when a robot misjudges its position in the formation, and begins handling joining and balancing as though it were somewhere else in the formation. Periodically, each of the entry point robots checks to see if its segment membership is valid. It also checks that its target is consistent with that segment membership. In the case of a mismatch, all neighbors robots occupying segments that should not exist at this location are instructed to drop formation and perform the merging operation, as described in subsection 3.2.2. This situation is illustrated in the right panel of Figure 3.5. at some point, robot A incorrectly assumed the role of leading entry point. This caused robot C to join in a place not consistent with the formation. Robot C will be instructed to leave the formation, and rejoin.

3.3 Implementation

I implemented my approach using a simulated environment, and simulated robots. Simulation allows for a large variation in the number of robots used, and the abilities of each robot. For my simulator, I have chosen to use Player/Stage Gerkey et al. [2001], because it supports the levels of heterogeneity that I required. It provided a simple interface to change the sensing and locomotion capabilities of the robots. The simulator provided all of the required functionality, and was not overly limiting. I employed an approach to robot control which is primarily Behaviour-based Arkin [1998]. Robots have multiple sets of behaviours, one set is selected based on the situation the robot finds itself in. This is a convenience for coding and debugging, and it is clear that this is simply a behaviour-based robot with certain behaviours that only apply situationally. Each behaviour represents a task that the robots are to perform. The tasks each have weight, determining their relative importance. Each generates a heading, and the weighted vector sum of these headings determines the actual heading of the robot.

3.3.1 Robot Internals

Each robot has a target robot, which it uses as a reference point. This target, combined with the robot's segment membership information allow the robot to maintain its position in the formation.

Each robot has four behaviours, each of which affects its final heading. The first such behaviour is the *formation keeping* behaviour. This behaviour is active only if the robot has a target. If so, the behaviour calculates the desired distance and angle to the target robot. The result of the behaviour is a vector that points towards this desired position. The next behaviour is *goal seeking*. As mentioned in section 1.3, robots have a shared destination, and move towards it. This is accomplished by giving each robot access to its absolute position and orientation information, and specifying the goal in absolute coordinates. In order to ensure that this global information does not contaminate the local sensor data, it is stored separately from all of the other sensor readings. It is used only to compute the goal vector. The other two behaviours are the two types of obstacle avoidance. *Robot obstacle avoidance* helps to avoid running into other robots, while *environmental obstacle avoidance* helps to avoid collisions between the robot and its environment. Each of these behaviours can be thought of as a motor schema in the approach described by Arkin [1998]. The behaviours used are similar to those used by Balch and Arkin [1998].

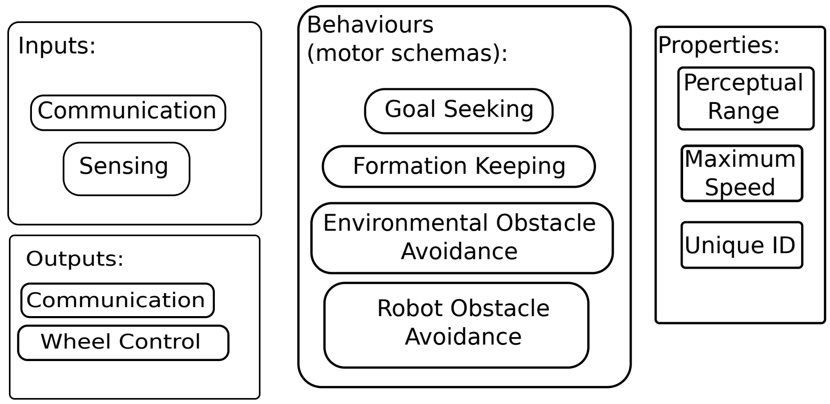


Figure 3.6: The internal components of a robot

Each robot has an internal unique ID, a defined maximum sensing range, and maximum wheel velocities. These are determined when the simulation is created, as physical properties of the robot. In order to achieve consistent performance with robots of different speeds, all wheel velocities are calculated using an assumption of a 2 m/s maximum velocity. These are then scaled to the actual capabilities of the robot. Figure 3.6 shows a visual representation of the components of a robot.

The inclusion of globally unique IDs is somewhat unrealistic. In a real world situation, it is possible that two robots would share an ID. The ID, however, was simply introduced as a way of simulating directed communication in the simulated environment. In a real-world environment, robots would require a mechanism for identifying others. This mechanism would likely consist of identifying combinations of visible features of others.

3.3.2 Sensing

Robot sensing is handled through calls to the simulator. Two sensing methods are used by each robot. The first is a laser scanner, for obstacle avoidance. My obstacle avoidance technique is described in subsection 3.3.3. The second is a fiducial scanner, giving position and identification of other nearby robots. This fiducial scanner and laser scanner have identical range, and are meant to simulate a single device, which can gather range data, and identify other robots. Both sensors have a 180 degree field of view. The maximum range of the sensors is adjustable, and can be varied for each robot individually.

3.3.3 Locomotion

There are three primary factors that control how fast a robot moves, and in which direction. The first is the vector sum of the evaluated behaviours. This is the only element that determines the direction of the robot. Two additional restrictions are placed on the speed of the robot. The first is whether or not a robot can perceive its target. If not, it sends out a message to the target robot. If the sending robot is both visible to its target and in front of its target, the target robot sends a message indicating this. Upon receiving such a message, the robot cuts its speed in half, to allow the target robot to catch up. Methods for this type of messaging are discussed in section 3.3.9. As noted in subsection 3.3.1 a robot cannot maintain its position in the formation without its target as a reference point. The final element of a robot's speed is the maximum speed of its neighbouring robots. This information is communicated periodically by neighbouring robots, as described in subsection 3.3.9. A robot will

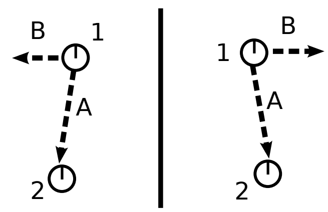


Figure 3.7: In both cases, robot 2 is attempting to avoid robot 1. Vector A represents the obstacle avoidance vector. vector B is the second right-angle vector, which helps robots move around each other.

not exceed the maximum speed of its neighbours. It is clear that doing so would make formation more difficult, if not impossible.

3.3.4 Obstacle Avoidance

Robots employ two methods of obstacle avoidance. Other robots within 2 metres of the robot are treated as repulsive forces. This is common in other behaviour-based control techniques Balch and Arkin [1998]. I found more success by also adding a small vector that points at a right angle to the sensed location of the obstacle robot. This helped to avoid situations where one robot would become trapped behind another, when it needed to go around. The direction of the right angle is determined by the sign of the angle assigned to the sensed robot. This process is illustrated in Figure 3.7. In this figure, robot 2 is attempting to avoid collision with robot 1. the addition of vector B allows robots to more easily move around each other.

Stationary objects are treated differently. To avoid stationary objects, robots employ a similar method to that used by the player/stage Gerkey et al. [2001] laser obstacle avoidance example. This technique looks for the largest unobstructed stretch

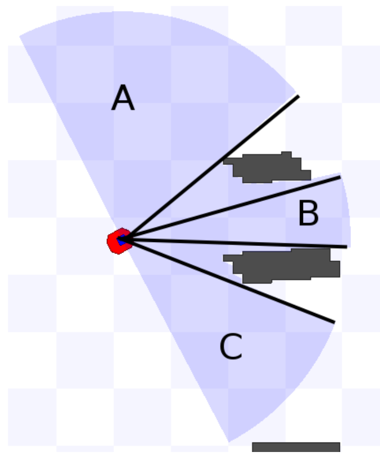


Figure 3.8: The robot in the picture can choose to move towards any of the open regions, denoted A, B and C respectively. region B is larger than 1 meter, and its centre point is closest to the robot's current heading. The robot moves towards region B, avoiding the obstacles.

of laser, and drives towards the middle of it. My technique uses a similar idea, but instead identifies the closest unobstructed stretch of the laser scan to the front of the robot. To be considered, the opening must be at least 1 metre in length. This ensures minimal deviation from course. The result of this obstacle avoidance is a vector pointing towards the desired open space. Figure 3.8 displays a situation where a robot has 3 possible paths around a group of obstacles. When determining which path is closer, the robot compares the distances of each of the centre points of the open areas to the current heading. In this particular case, the robot would choose path B, since it is large enough, and its centre point is closest to the robot's current heading.

In the case that the robot strikes another object, the simulator sets a flag indicating this. Because the newer versions of the stage simulator no longer support bumper devices, this signal was used as a substitute for them. If the collision flag is set, the robot reverses course for one second, then continues moving forward.

3.3.5 Internal Representation of the Formation

Each robot has in internal representation of the formation. This is at the core of how robots maintain their place in the formation.

One use of this internal representation is determining a robot's segment membership. We discussed in section 3.2.1 that robots can be members of multiple segments. This is represented internally by a bit vector, indicating segment membership. I will refer to this bit vector as a robot's *segment identifier*. This has two main advantages. First, it allows robots to quickly use a logical and operation to see if they share a segment. Second, it allows efficient computation of whether or not a robot is a member of a given segment, without requiring a search operation. The disadvantage of this approach is that it places an upper limit on the number of segments that a formation can have. In my implementation, the bit vector is stored in a 32 bit integer. This limits formations to having 32 segments. This is sufficient for all formations that I have implemented, but it could be easily expanded to 64 segments by using a 64 bit integer.

For formations such as the one pictured in figure 3.9, robots need a mechanism for determining which segments should terminate at each entry point. To ensure that any robot can pick up the role of entry point at any time, becoming an entry point also involves a re-evaluation of a robot's segment identifier. To determine its segment identifier, a robot first examines the segment identifiers of all of its neighbors. Next, it checks for one of four possible conditions. If the robot has no target, it becomes the leading entry point robot in its segment. if a robot has only one neighbour, it becomes the trailing entry point of its segment. These two cases are intuitive, and follow naturally. The next case allows for becoming an entry point robot in the middle of a formation. If the robot has at least 2 neighbours, and shares no segments with its target, the robot is the leading entry point of its segment. Finally, if none of the above cases is true, then the robot simply maintains its segment identifier. The three cases resulting in a change in segment identifier can be illustrated using Figure 3.9. In this figure, robot A should have segment identifier 110, robot B should have a segment identifier of 011 and robot C should have a segment identifier of 001. In our first hypothetical situation, Robot A has joined the formation, and initially has a segment identifier of 010. In this case, robot A has no target. According to the rules described in this section, A should become the leading entry point for its segment. It examines its representation of the formation, finding that the leading entry point of segment 2 is also the trailing entry point of segment 1. Robot A then assumes the segment identifier that matches this position, which is 110, as expected. Our second hypothetical situation involves robot B. Robot B has joined the formation, and initially is given a segment identifier of 001. Robot B's target is in a different segment, and there is no overlap between their current segment identifiers. Examining the rules in this section, Robot B must then become the leading entry point in its segment. The leading entry point of 3 is also the trailing entry point of segment 2, giving a segment identifier of 011, as expected. Finally, robot C has only one neighbour, making it the trailing entry point of segment 3. Its segment identifier remains unchanged, at 001.

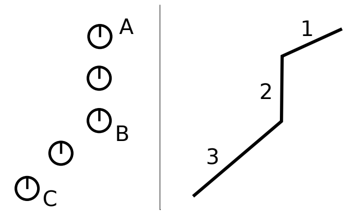


Figure 3.9: Segment identifier determination for a 3 segment formation. The left panel illustrates the robots making up segments 2 and 3. The right panel illustrates the shape of the desired formation, with each segment numbered.

Finally, the internal formation representation is used for joining and balancing, as described in sections 3.2.2 and 3.2.5 respectively. Segment identifiers are used to manage the counting of segments, and are used as a part of balancing and joining operations.

3.3.6 Robot Capabilities and Environment

Robot heterogeneity has two dimensions within the system. These dimensions are sensing ability, and locomotion ability. When a robot is created, its strength level in each of these two areas must be specified. These are specified in the form of a maximum wheel velocity, and a maximum sensing distance.

The virtual world in use also varies from trial to trial. Worlds are classified as easy, moderate or difficult. An easy world has no obstacles. A moderate world has large obstacles which are widely spaced. A difficult world has many small and closely spaced obstacles. The specific moderate and difficult worlds in use are pictured in Figure 3.10.

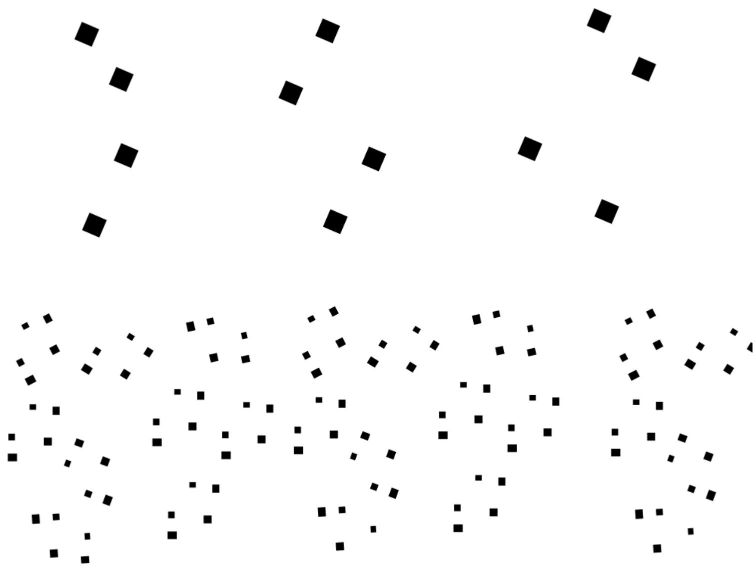


Figure 3.10: Above is a sample of the moderate world, below is a sample of the difficult world.

3.3.7 Formation Membership

Each distinct formation is identified by the ID of the robots that originated it, and the time of creation. This number is then propagated along the formation. A robot can request another robot's formation ID at any time. This formation ID is essential for properly merging two formations, and for determining when not to merge formations. The ID is obtained by concatenating the IDs of the two forming robots together.

Periodically, each robot sends out heartbeat messages to all of its neighbours. These messages contain status information, including formation and segment membership. If any of this information indicates that this robot should not be a neighbour, the sending robot is instructed to perform the merging operation described in subsection 3.2.2. Possible indications that two robots should not be neighbours are differing formation IDs and belonging to segments that do not intersect.

As time goes on, Robots may lose contact with their neighbours. Two methods are used to ensure that neighbours are still present, and have not changed formation or segment membership. The first method is the heartbeat message mentioned earlier in this section, the second is through sensing. If we have neither sensed a neighbour, nor received any communication from it in 5 seconds, we assume that it has died, or moved to another part of the formation. It is then dropped as a neighbour. If the robot is still still willing and capable of participating in the formation, it may rejoin later. This timeout seems short, but it ensures that formations quickly adapt to lost members. If a robot is dropped from the formation erroneously, it can rejoin.

3.3.8 Counting

In order to properly maintain counts of robots in each of their segments, entry point robots periodically send out counting messages. These messages are to be relayed along the segment until an entry point robot is reached. Each robot increments the count as it forwards the message. Upon reaching an entry point robot, the count is incremented one final time. Finally, the message is relayed back along the segment. Each robot updates its segment count to reflect the value in the message. In the case where the message is not relayed properly, robots simply wait. The broken communications link results in a removal of a neighbour, and a change in entry point robots. At this point, lines of communication between the two entry point robots should be clear.

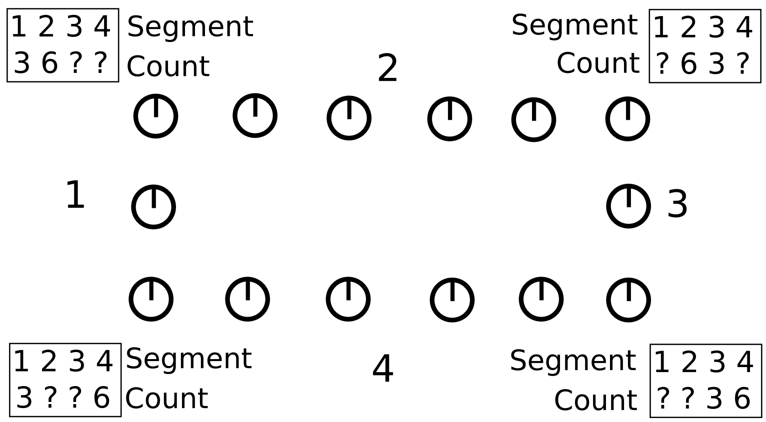


Figure 3.11: A representation of what each of the entry point robots knows about the counts of segments in the formation. The information for each entry point robot is in the corner closest to it.

The counting operation gives robots information on the counts of one or more segments in the formation. Figure 3.11 illustrates the segment counts known by each of the entry point robots in the formation.

In order to ensure that the formation is stable before beginning balancing, entry point robots wait until at least two counting messages with the same count are received for each neighboring segment. Any time a counting message arrives with a different count than what is stored by the robot, balancing is delayed. This is done to ensure that the balancing and joining operations do not interfere.

3.3.9 Communication

A large portion of the logic involved in establishing and maintaining a formation is encoded into a variety of different types of messages, which can be sent to any robot within communications range.

To facilitate inter-robot communication, each robot ID maps to an IP address and port combination. These mappings are known by all robots ahead of time. This is necessary to allow robots to intercommunicate. This piece of global information would not be needed in a purely physical implementation of this system. It would be handled implicitly by the wireless transceivers of the individual robots. Messages are also forwarded to a messaging server, to allow them to be logged.

Each message type has an associated timeout. The timeouts are tunable, but are generally between 1 and 2 seconds. Once a robot has sent a message of a given type, it cannot send another message of that type until the timeout has elapsed. This is designed to keep the communication channels clear, and help to reduce the number of lost messages. The timeouts can also serve as a guideline on when to perform certain actions. For example, if a robot has recently transmitted a join request, it should not be able to receive join requests from others. Generally speaking, a robot may not perform any action that would invalidate an outstanding message. e.g. accepting a join request in the middle of a balancing operation.

As the formation structure is constantly changing, it is possible that a message could be relayed to a robot that has already seen it. This is a temporary condition that can arise when robots are changing targets, segments or doing balancing or joining. For this reason, each robot keeps track of the last 10 messages of each type that it has seen. Forwarded messages are given a unique identifier, which causes them to be discarded if duplicates are seen.

Due to the nature of network communication, it is possible that messages in transit can become irrelevant or even incorrect before reaching their destination. For this reason, all messages that can cause a change in the formation are timestamped. These timestamps are compared to the time a robot joined its current formation. Messages that predate the joining are ignored. Messages that occur a short time after changing targets, segments or formations are also ignored.

The different messages are as follows:

Join Request Sent from a robot that is not in a formation, upon encountering another robot. The response indicates the formation of the remote robot. If the remote robot is an entry point, the robot is given a vector to the point where it should join. Otherwise, the ID of the next potential entry point is returned.

Segment Change Message Sent from a robot to its immediate neighbours. This message instructs the recipient to change its segment membership to the desired segment. This message is used primarily during the balancing operation.

Target Change Message Sent from a robot to its immediate neighbours. This message instructs the recipient to use the desired robot as a reference point. This message is used as part of joining, merging and balancing.

Target Selection Message Sent from a robot following a change of its target. Informs the target that it is now being used as a reference point by the sender. This allows the receiving robot to keep a more accurate listing of its neighbours.

Balancing Message This message is sent to the source robot in a balancing operation. The balancing entry point robot can only communicate with its immediate neighbours. This message serves to make the balancing robot and the next robot in the source segment aware of each other.

Visibility Request An robot sends this message to its target, if the target is not visible. The target robot, receiving this message responds with an affirmative if the requesting robot is visible to it.

Counting Message This message is regularly sent from the entry point to the first robot in a segment. The message is forwarded along the segment until it reaches another entry point. the count is incremented at each hop. Upon receipt by another entry point, the message is then returned, fully populated with counts. Potential loss of the message along the way is the reason why we consider all counts to be approximate.

Formation Information Request This message is the precursor to merging. It allows two entry point robots to determine if they belong to the same formation, and if not, which formation is larger. Encoded in the message are the formation ID and formation size of the sending robot's formation. This message generates a response including the formation ID and formation size of the receiving robot's formation. Following this exchange, the entry point robots can initiate merging. If the two robots are members of the same formation, and they have one segment in common, they become neighbours.

Split and Join Request This message instructs the receiver to leave its current formation, and attempt to join the formation containing the specified target robot. This message is then forwarded to all neighbouring robots. This message is used to implement the merging process described in section 3.2.2

3.4 Summary

This chapter described the core of my approach and how it was implemented. It explained my representation of a formation as a series of interconnected line segments. Also discussed was the idea of limiting points of entry to the formation. The approach section described at a high level, all of the requirements for joining and maintaining a formation. The implementation section built on this by explaining the underlying mechanisms that drive the approach.

Chapter 4

Evaluation

4.1 Introduction

I chose to evaluate my approach in simulation for a number of reasons. The large number of trials needed was one reason. Another reason was the ability to quickly and easily modify the sensing and locomotion capabilities of the robots. Finally, I chose to evaluate my approach in simulation because of the ability to precisely repeat trials.

4.2 Metrics for Comparison

The conditions under which my approach operates significantly exceed others, and thus it is difficult to make a direct comparison in the same environments. However, I employ two measures commonly used by others, so that overall performance relative to prior approaches can be discussed.

First, I define the error in the largest formation. To compute this, it is required to calculate the optimal configuration of robots. Since formations can have a variable number of members, it is necessary to calculate the optimal lengths of each segment. The relative lengths specified in the formation definition are used to determine the set of optimal segment lengths for a given number of robots. Let T be the total number of robots in the formation, L be the sum of all relative lengths in the formation and L_n be the relative length of segment n. For each segment n, the optimal length O_n can be calculated as follows:

$$O_n = \frac{L_n}{L} * T$$

Let the actual number of robots in segment n be A_n. The error for each segment is then calculated as $E_n = |O_n - A_n|$. The *formation error* is the sum of the error values for each segment in the formation.

Next, I define the time to establish a formation. This is the time when the formation error remains constant for 10 seconds, and all robots are participating in the same formation. This indicates that no further balancing will occur, and that the formation is relatively stable. The formation may not be error-free at this point, but the level of error at this time will give a good indicator of how well the technique achieves the final formation.

The statistics relating to the largest formation are a good way to evaluate trials where all robots achieve a single formation. For other cases, it is important to examine some other metrics. The first such metric is the maximum number of robots in formation over the course of the trial. This is more relevant than the size of the largest formation in cases when the simulation time runs out. This is because formations

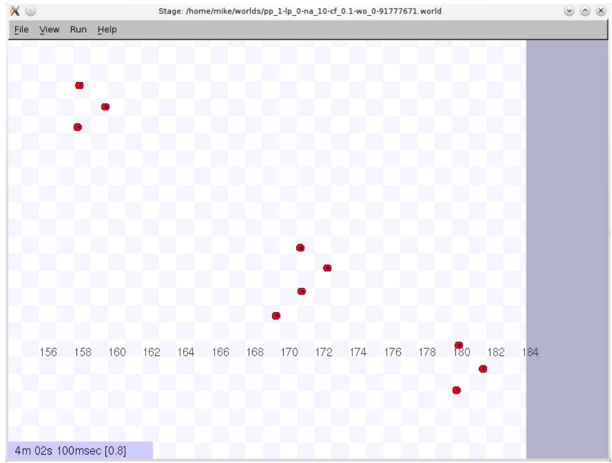


Figure 4.1: A group of robots, creating three separate formations.

may be in the middle of a merging operation. In such cases, it is better to look at the maximum number of robots in formation, and the formation error at that time.

In cases with large numbers of robots, it is possible that robots get separated, and are entirely out of perceptual range, but are otherwise in correct formations. An example of such a situation is illustrated in Figure 4.1. In such cases, the number of robots participating in any formation and the number of robots in the largest formation give a good understanding of what is going on. If there are many robots participating in formations, but the size of the largest formation is small, it indicates that groups of robots have moved into a series of smaller formations, instead of one large one. A small number of robots participating in formations indicates that a large number of robots have been individually separated from the rest of the group over the course of the trial.

For each experimental trial run, I recorded the final formation error, time to establish a formation, number of robots in the largest formation, maximum number of robots in the largest formation, maximum number of robots participating in any formation and the number of distinct formations at trial end. All of these variables were sampled once per second. I also calculated the number of trials resulting in all robots in a single formation.

4.3 Experimental Domain

I evaluated my approach using three different domains. Each of these is represented by a world file, as accepted by the player/stage Gerkey et al. [2001] simulation package. These domains have varying degrees of obstacle cover. The default domain used in the majority of experiments is empty, and contains no obstacles. The domains containing obstacles are illustrated in Figure 3.10 in Chapter 3.

4.4 Experiment Overview

I performed the simulated trials using a modified factorial design. I split the tests into 6 groups, testing the effects of sensing and locomotion ability on scalability, adaptability and robustness separately. The first three groups measured the effect of varying sensing capabilities as number of robots, chance of failure and world were varied respectively. The three remaining groups measured the effect of varying locomotion ability as number of robots, chance of failure and world were varied respectively. All trials consisted of formation movement in a known direction. A trial is considered to have concluded when a formation has been established for 5 seconds, or after a time limit of 5 minutes has elapsed.

For these trials, I chose a V formation, with two segments of equal relative length. Distance functions for both segments were set to a constant of 2 metres. Angle functions were also constant, but one segment used 45 degrees while the other segment used -45 degrees. I expect that the results for a V formation will generalize well to other types of formations. The only real difference between a V and a more complex shape is the amount of balancing that needs to be done to achieve a correct formation. Joining and merging operations would be handled identically for any formation.

Robots were initially distributed on a 6 metre by 6 metre grid. Robot orientation was random. Robots were positioned on the grid points of a 6 metre square grid, with separation of 1 metre.

Each trial involved probabilistic robot failure. Every 30 seconds, a server generates a random number in the range [0..1]. If this number is less than the probability of failure in the current trial, a randomly selected robot is instructed to fail. This robot will no longer move or communicate during this trial.

I tested scalability by examining the time to establish formation and the per-robot error, as the number of robots was increased. Success was measured by the relationship between number of robots and time taken to establish formation. A linear relationship with stable error was considered to be a good result. For these trials, I used a fixed world and probability of failure, and varied the number of robots.

I tested adaptability by examining the error rate and time to establish formation as the environment difficulty increased. I expected to see a fairly stable rate of error, and an increase in time to establish formation roughly proportional to the additional obstacle cover. For these trials, the number of robots and probability of failure remained fixed, while the world in use was varied.

Finally, robustness was measured by the maximum number of robots in formation and the number of robots in the formation when the trial ended. My expectation was thats both numbers would drop off predictably as probability of failure increased. The more shallow the descent, the higher the degree of robustness. For these trials, number of robots and world remained fixed, while probability of failure was varied.

I used the following factors:

Number of Robots In order to test scalability I tested with a small, moderately sized and large group. Populations of 5, 10 and 15 sufficiently illustrated the capabilities of the system.

Sensing ability I started with a baseline of all moderate robots, using identical sensor profiles. I then studied the effect of increasing the number of weak robots, first to 25% then to 50%. Next, I removed the weak robots and tested the introduction of strong robots into the group of moderates, again, using 25% and 50%. Finally, I determined the effect of simultaneously increasing the levels of strong and weak robots. First increasing strong and weak to 25%, then increasing both to 50%. This gives a total of seven levels for this factor.

Locomotion Ability I used the same 7-level approach as described under sensing ability.

World This factor had three levels, each corresponding to one of the difficulty levels. At each level, the system randomly selected a world of the given difficulty level.

Probability of failure The levels 0, 0.1 and 0.3 illustrated the capabilities of the system.

I performed 50 repetitions of each factor combination in each group, resulting in a total of 4550 simulation runs. These simulations were run in parallel on 50 c1.medium computing instances on Amazon's EC2 cloud computing service.

4.5 Results

4.5.1 Interpreting the Charts in This Chapter

The charts in this chapter include many references to sensing profiles, and locomotion profiles. Tables 4.1 and 4.2 describe the sensing and locomotion profiles respectively. Each profile defines a percentage of the robots in the trial which have weak, moderate or strong locomotion or sensing abilities. Each chart represents averages over 50 trials, with error bars indicating the standard deviation of the data. In cases with low standard deviation because of low numbers of data points, the standard deviation is calculated across sensing and locomotion profiles, instead of individually for each profile.

Several of the metrics presented in the sections that follow are collected at the end of the trial. I will refer to these metrics as the *end of trial metrics* throughout the sections that follow. These include formation error, number of robots in the final formation, time to achieve formation, and number of distinct formations at the end of the trial. If the approach has not yet formed a single formation, this leaves open the possibility that the formation is in a number of different states. The formation may have just finished a merge, in which case, formation error would be abnormally

Sensing Profile	Weak	Moderate	Strong
0	0%	100%	0%
1	25%	75%	0%
2	50%	50%	0%
3	0%	75%	25%
4	0%	50%	50%
5	25%	50%	25%
6	50%	0%	50%

Table 4.1: This table describes the percentage of robots with weak, moderate, or strong sensing capabilities for each sensing profile.

high. The formation may be in the process of merging, meaning that the number of formations, and the number of robots in the final formation would be artificially low. This explains the very large error bars on these metrics in this chapter.

The metrics dealing with formation size assume that the minimum size of a formation is 2 robots. Single robots are not considered to be in formation.

The time to establish formation is only recorded in cases when all robots are participating in a single formation. In other cases, this metric is not used, and is effectively infinite. Many of the charts show no bar for time to establish formation. This means that there was not a single trial where all robots ended in one formation.

Number of trials ending in a single formation is also different from the other graphs. Instead of being an averaged value across multiple runs, it is instead a count of the number of trials when a single formation consisting of all robots was formed. This value is primarily used to give context to some of the other metrics, and explain some of the reasons for the large error bars.

Locomotion Profile	Weak	Moderate	Strong
0	0%	100%	0%
1	25%	75%	0%
2	50%	50%	0%
3	0%	75%	25%
4	0%	50%	50%
5	25%	50%	25%
6	50%	0%	50%

Table 4.2: This table describes the percentage of robots with weak, moderate, or strong locomotion capabilities for each locomotion profile.

4.5.2 Scalability Trials

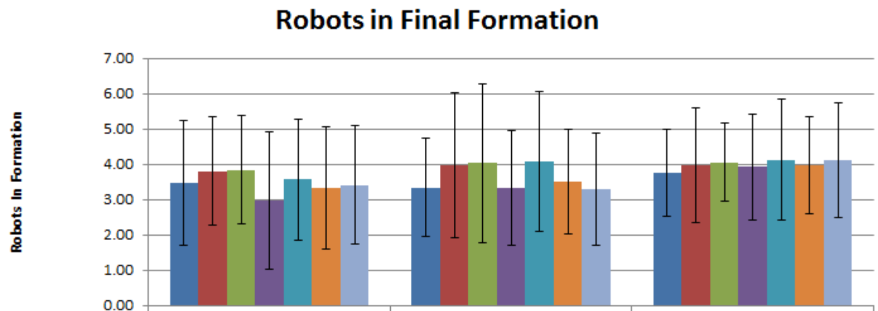
Scalability trials were performed with no chance of robot failure, in an easy world. In a highly scalable approach, I would expect to see an increase in the number of robots in formation. This was the case, although they were not always in the same formation.

Varying Sensing

I observed little difference in performance between the different sensing profiles in any of the collected metrics. This indicates that the scalability of the approach is independent of heterogeneity in robot sensing. Figure 4.2 indicates the number of robots in formation at the end of each trial. It indicates that the number of robots in the final formation varies between 3 and 4 for all scalability levels. This seems like poor performance before looking at Figure 4.8. This figure shows a clearly increasing trend. As the number of robots increases, so does the number of distinct formations. Formation error also increases slightly as the number of robots increases. Figure 4.3 illustrates this increase. The increased error is likely as a result of trials with 10 or

15 robots, as the majority of such trials did not result in one formation, but instead resulted in many. Many such trials were stopped before the completion of a balancing operation. Examining Figure 4.5 shows a strong correlation between the decrease in the number of complete single formations, and the rise in formation error. This is evidence for the claim that balancing had not happened yet in some of these trials. Figure 4.6 demonstrates a strong relationship between the number of robots in the trial, and the maximum number of robots to be in formations at the same time. This provides further evidence to support the fact that multiple formations are developing, as illustrated in Figure 4.1 and described in section 4.2. Figure 4.7 also indicates a clear relationship between the number of robots, and the maximum size of the largest formation. This provides further evidence of the scalability of my approach. As mentioned in Section 4.5.1, the error bars for the end of trial metrics are quite large. Looking at the raw data, it is clear that a large amount of the standard deviation can be accounted for by trials that were interrupted during a merging or balancing operation. If these trials had continued longer, or been cut off sooner, the standard deviation in these metrics would have been dramatically lower.

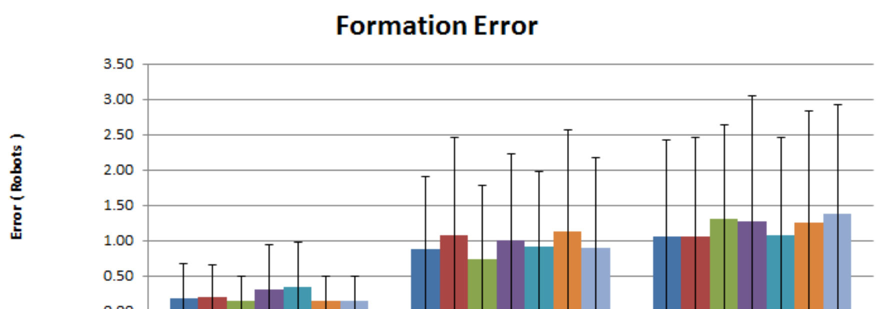
Finally, I examine the time to achieve formation, as described by Figure 4.4. The figure shows no major difference between 5 and 10 robots. As noted earlier in this section, this may not be entirely valid, as few trials of 10 or 15 robots resulted in all robots in a single formation.



	5	10	15
Sensing Profile 0	3.46	3.33	3.75
Sensing Profile 1	3.80	3.98	3.98
Sensing Profile 2	3.84	4.02	4.04
Sensing Profile 3	2.96	3.32	3.92
Sensing Profile 4	3.56	4.08	4.12
Sensing Profile 5	3.32	3.49	3.96
Sensing Profile 6	3.40	3.30	4.12

Robots

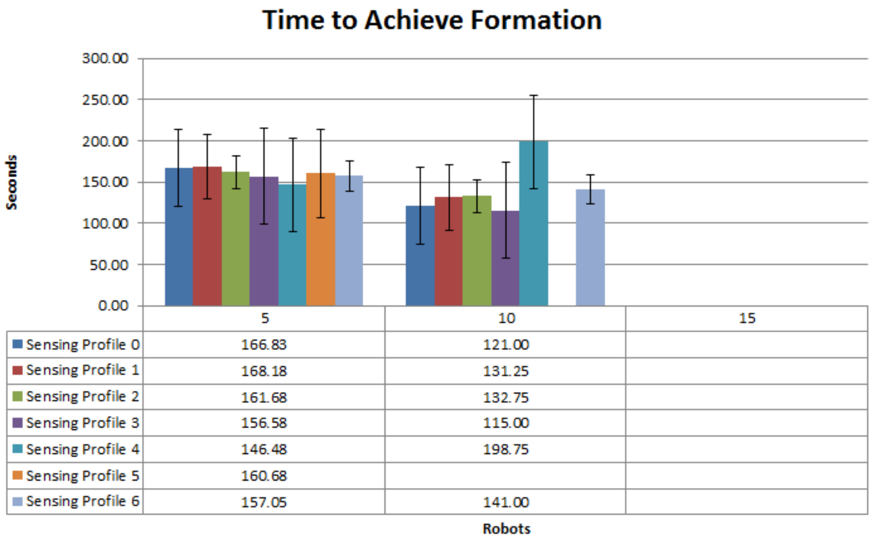
Figure 4.2: The relationship between the number of robots, and the number of robots in the largest formation at the end of the trial.



	5	10	15
Sensing Profile 0	0.18	0.88	1.05
Sensing Profile 1	0.20	1.08	1.06
Sensing Profile 2	0.14	0.74	1.30
Sensing Profile 3	0.30	1.00	1.26
Sensing Profile 4	0.34	0.92	1.08
Sensing Profile 5	0.14	1.12	1.24
Sensing Profile 6	0.14	0.90	1.38

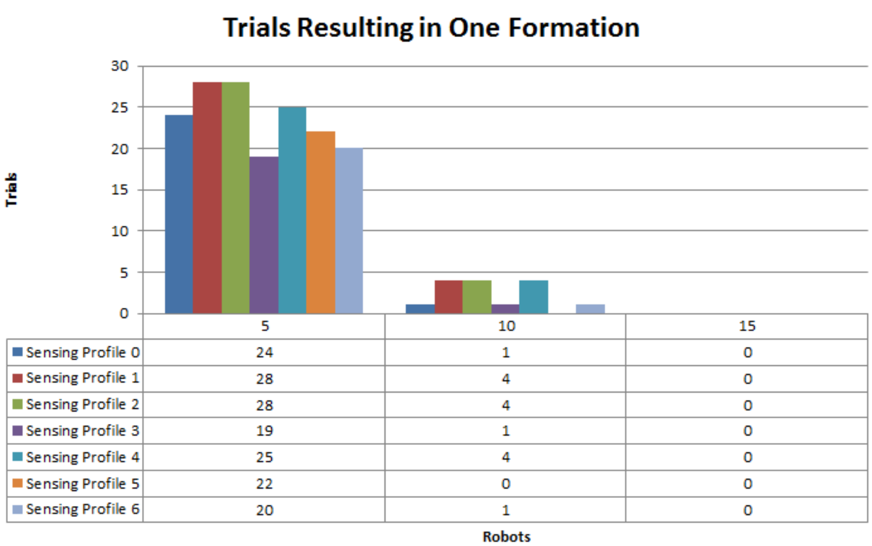
Robots

Figure 4.3: The relationship between the number of robots, and the formation error at the end of the trial.



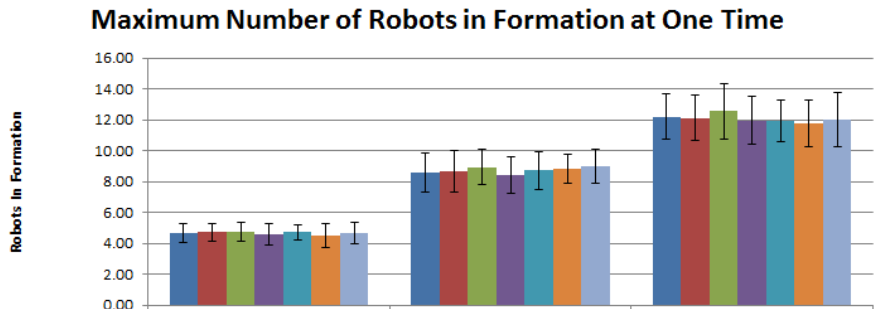
	5	10	15
Sensing Profile 0	166.83	121.00	
Sensing Profile 1	168.18	131.25	
Sensing Profile 2	161.68	132.75	
Sensing Profile 3	156.58	115.00	
Sensing Profile 4	146.48	198.75	
Sensing Profile 5	160.68		
Sensing Profile 6	157.05	141.00	

Figure 4.4: The relationship between the number of robots and the amount of time it takes to establish a single formation.



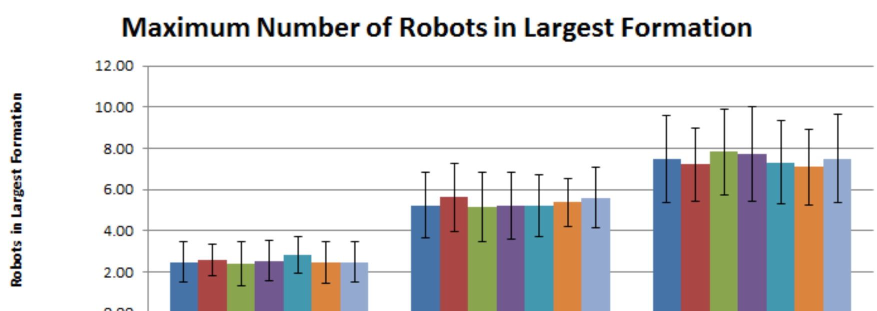
	5	10	15
Sensing Profile 0	24	1	0
Sensing Profile 1	28	4	0
Sensing Profile 2	28	4	0
Sensing Profile 3	19	1	0
Sensing Profile 4	25	4	0
Sensing Profile 5	22	0	0
Sensing Profile 6	20	1	0

Figure 4.5: The number of trials resulting in a single formation at the end as the number of robots changes.



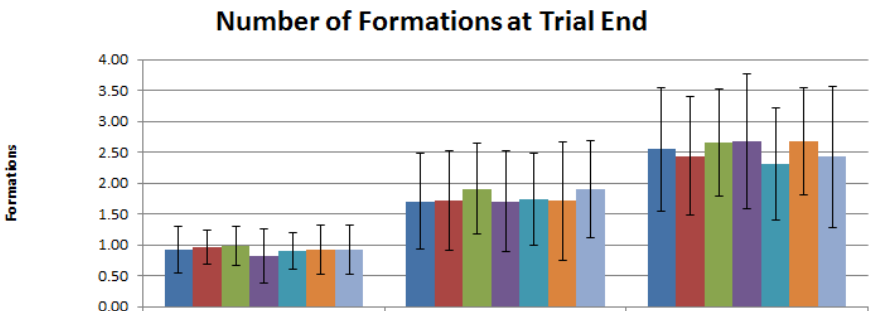
	5	10	15
Sensing Profile 0	4.67	8.58	12.21
Sensing Profile 1	4.73	8.64	12.12
Sensing Profile 2	4.74	8.94	12.54
Sensing Profile 3	4.56	8.42	11.96
Sensing Profile 4	4.70	8.70	11.94
Sensing Profile 5	4.48	8.82	11.78
Sensing Profile 6	4.64	8.96	12.02

Figure 4.6: The maximum number of robots to be participating in formations at the same time as the number of robots changes.



	5	10	15
Sensing Profile 0	2.48	5.23	7.47
Sensing Profile 1	2.57	5.62	7.20
Sensing Profile 2	2.40	5.14	7.82
Sensing Profile 3	2.52	5.20	7.72
Sensing Profile 4	2.84	5.20	7.30
Sensing Profile 5	2.44	5.37	7.08
Sensing Profile 6	2.48	5.58	7.50

Figure 4.7: The maximum number of robots in the largest formation as the number of robots changes.



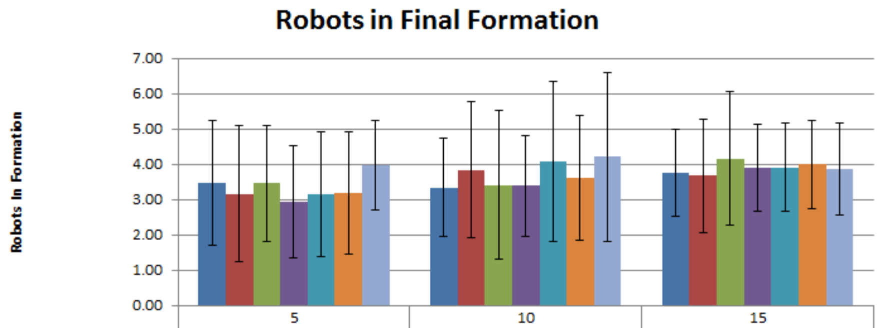
	5	10	15
Sensing Profile 0	0.91	1.70	2.55
Sensing Profile 1	0.96	1.72	2.44
Sensing Profile 2	0.98	1.90	2.66
Sensing Profile 3	0.82	1.70	2.68
Sensing Profile 4	0.90	1.74	2.30
Sensing Profile 5	0.92	1.71	2.67
Sensing Profile 6	0.92	1.90	2.42

Figure 4.8: The number of distinct formations at the end of the trial, as the number of robots changes.

Varying Locomotion

All of the locomotion profiles performed similarly on all of the metrics that I examined. This is a sign that heterogeneity in robot locomotion does not play a significant role in the scalability of the approach. Figure 4.9 indicates the number of robots in formation at the end of each trial. It illustrates that the number of robots in the final formation varies between 3 and 4 for all scalability levels. This seems like poor performance before looking at Figure 4.15. This figure shows a clearly increasing trend. As the number of robots increases, so does the number of distinct formations. Formation error also increases slightly as the number of robots increases. This can be seen in Figure 4.10. This increase can be attributed to the fact that most of the trials with 10 or 15 robots did not result in all robots in a single formation, and therefore, were possibly stopped before the completion of a balancing operation. Figure 4.12 shows that none of the trials involving 15 robots ended with all robots in one formation. Examining Figure 4.12 Shows a correlation between the decrease in the number of complete single formations, and the rise in formation error. Figure 4.13 demonstrates a strong relationship between the number of robots in the trial, and the maximum number of robots to be in formations at the same time. This provides further evidence to support the fact that multiple formations are developing, as illustrated in Figure 4.1 and described in section 4.2. Figure 4.14 also indicates a clear relationship between the number of robots, and the maximum size of the largest formation. This provides further evidence of the scalability of my approach. As mentioned in Section 4.5.1, the large error bars on the end of trial metrics are due to trials that were interrupted in the middle of a balancing or merging operation.

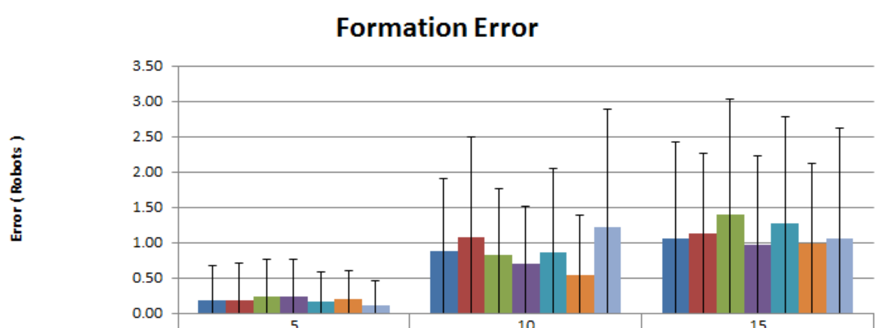
Finally, I examine the time to achieve formation, as described by Figure 4.11. The figure shows no major difference between 5 and 10 robots. As noted earlier in this section, this may not be entirely valid, as few trials of 10 or 15 robots resulted in all robots participating in a single formation.



	5	10	15
Locomotion Profile 0	3.46	3.33	3.75
Locomotion Profile 1	3.16	3.84	3.67
Locomotion Profile 2	3.46	3.40	4.16
Locomotion Profile 3	2.94	3.38	3.90
Locomotion Profile 4	3.14	4.06	3.90
Locomotion Profile 5	3.18	3.60	4.00
Locomotion Profile 6	3.96	4.20	3.86

Robots

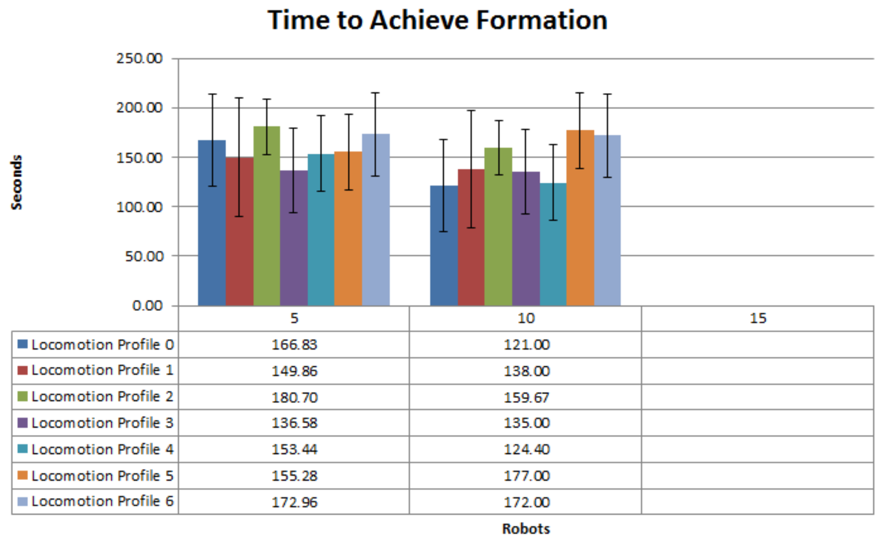
Figure 4.9: The relationship between the number of robots, and the number of robots in the largest formation at the end of the trial.



	5	10	15
Locomotion Profile 0	0.18	0.88	1.05
Locomotion Profile 1	0.18	1.08	1.12
Locomotion Profile 2	0.24	0.82	1.39
Locomotion Profile 3	0.24	0.70	0.96
Locomotion Profile 4	0.16	0.86	1.27
Locomotion Profile 5	0.20	0.54	0.98
Locomotion Profile 6	0.10	1.22	1.06

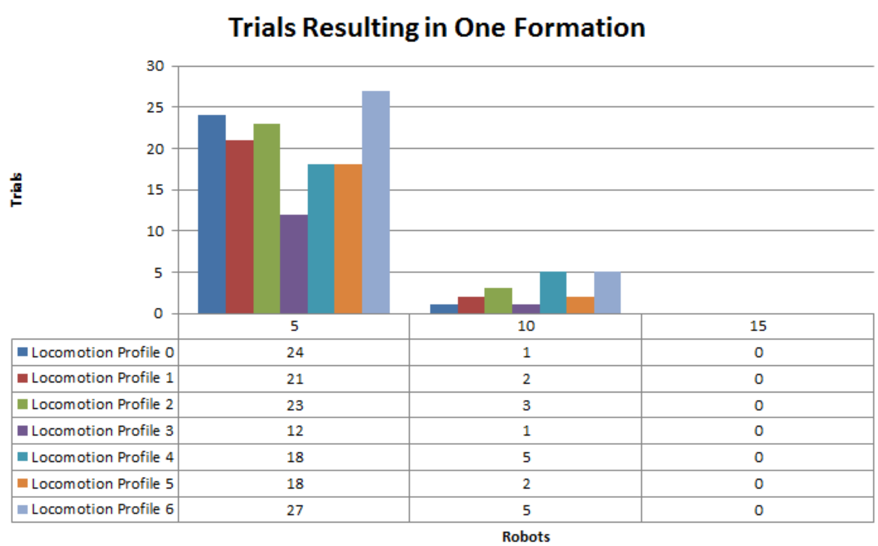
Robots

Figure 4.10: The relationship between the number of robots, and the formation error at the end of the trial.



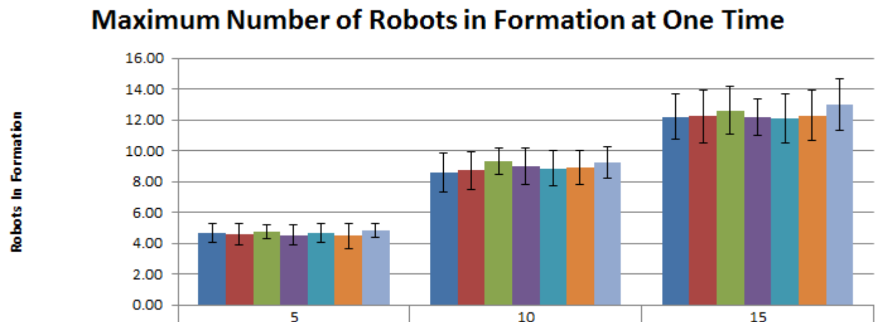
Robots	5	10	15
Locomotion Profile 0	166.83	121.00	
Locomotion Profile 1	149.86	138.00	
Locomotion Profile 2	180.70	159.67	
Locomotion Profile 3	136.58	135.00	
Locomotion Profile 4	153.44	124.40	
Locomotion Profile 5	155.28	177.00	
Locomotion Profile 6	172.96	172.00	

Figure 4.11: The relationship between the number of robots and the amount of time it takes to establish a single formation.



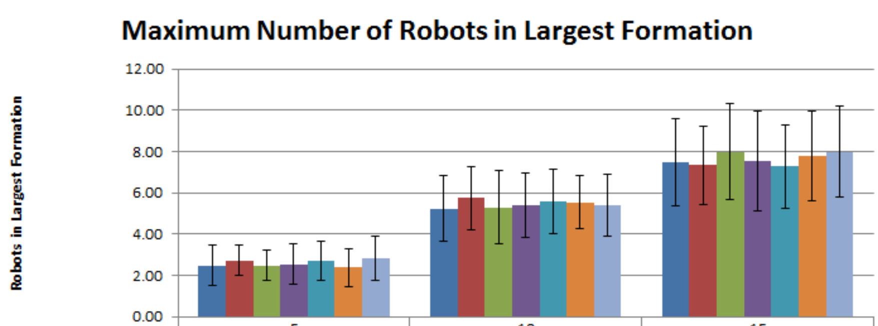
Robots	5	10	15
Locomotion Profile 0	24	1	0
Locomotion Profile 1	21	2	0
Locomotion Profile 2	23	3	0
Locomotion Profile 3	12	1	0
Locomotion Profile 4	18	5	0
Locomotion Profile 5	18	2	0
Locomotion Profile 6	27	5	0

Figure 4.12: The number of trials resulting in a single formation at the end as the number of robots changes.



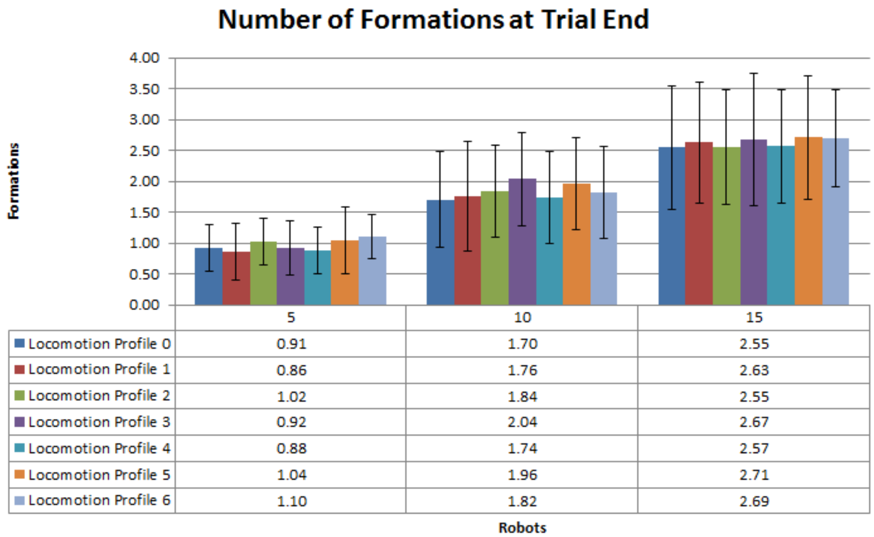
	5	10	15
Locomotion Profile 0	4.67	8.58	12.21
Locomotion Profile 1	4.58	8.72	12.22
Locomotion Profile 2	4.74	9.30	12.61
Locomotion Profile 3	4.52	8.98	12.16
Locomotion Profile 4	4.66	8.84	12.12
Locomotion Profile 5	4.46	8.92	12.27
Locomotion Profile 6	4.80	9.24	13.00

Figure 4.13: The maximum number of robots to be participating in formations at the same time as the number of robots changes.



	5	10	15
Locomotion Profile 0	2.48	5.23	7.47
Locomotion Profile 1	2.72	5.74	7.33
Locomotion Profile 2	2.48	5.28	7.98
Locomotion Profile 3	2.52	5.38	7.53
Locomotion Profile 4	2.70	5.58	7.27
Locomotion Profile 5	2.36	5.54	7.76
Locomotion Profile 6	2.82	5.40	7.98

Figure 4.14: The maximum number of robots in the largest formation as the number of robots changes.



	5	10	15
Locomotion Profile 0	0.91	1.70	2.55
Locomotion Profile 1	0.86	1.76	2.63
Locomotion Profile 2	1.02	1.84	2.55
Locomotion Profile 3	0.92	2.04	2.67
Locomotion Profile 4	0.88	1.74	2.57
Locomotion Profile 5	1.04	1.96	2.71
Locomotion Profile 6	1.10	1.82	2.69

Figure 4.15: The number of distinct formations at the end of the trial, as the number of robots changes.

4.5.3 Robustness Trials

For robustness trials, I used 10 robots, and no changes to the world in use. Using 5 would not allow for the proper testing of higher chances of failure. Losing 2 robots in a 5 robot simulation would largely invalidate the trial, since 3 robot formations are much easier to create and maintain. The three combinations that I tested were no chance of failure, a 10% chance of failure with a maximum of one failed robot and a 40% chance of failure with a maximum of two failed robots. Overall, robustness trials were quite successful. Ideally, there would be no change in any of the metrics as chance to fail increased. This is essentially what I observed. Over the course of the robustness trials, variations in sensing and locomotion profiles produced very different numbers of trials resulting in all robots in a single formation. I did not notice any pattern in this variability as locomotion and sensing profiles were varied.

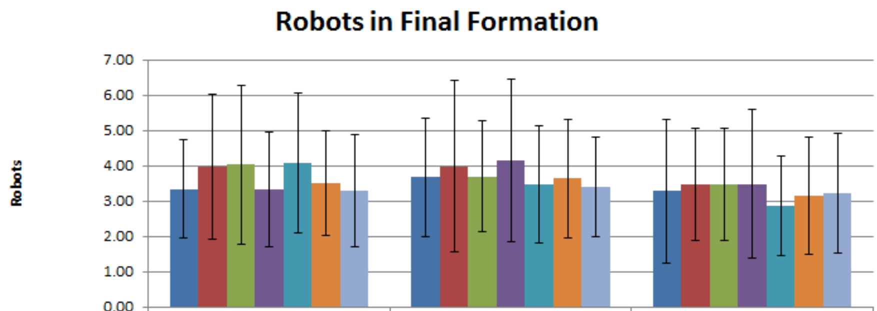
Varying Sensing

First, I will examine how variations in robots' sensing abilities interacted with the variation of the chance to fail, and maximum number of possible failures. Overall the variations in sensing were handled well by the underlying mechanisms of my approach. Changes in sensing had little impact on any of the error metrics that are reported in this section. This is positive, as it provides support for the fact that my approach is useful in the case of heterogeneous sensing.

Figure 4.16 seems like a condemnation of my approach. It shows that in a 10 robot trial, between 3 and 4 robots were in the largest formation at the end of the average trial. When looking at this number a bit more deeply, and combining it with the results in table 4.19 it is clear that most of the trials in question did not end with a proper formation. This means that they were cut off at some arbitrary point. They may have been in the middle of a merging operation. Two better measurements to use in such a case are the results in figures 4.20 and 4.21. These graphs clearly show that over the course of a trial, on average 8 or more of the 10 robots in a trial participate in a formation. Of those, 4 to 5 participated in the largest formation. This indicates that there are likely several distant formations forming throughout the trials. This exact problem is illustrated in Figure 4.1 and described in section 4.2.

The time to achieve formation, described by Figure 4.18 seems to suggest that robot failure chance has little effect on the time to establish a formation, nor does the level of robot sensing. When looking more closely, and examining the data in Figure 4.19, it becomes clear that there is too little data to conclusively say this.

Figure 4.17 shows promising results. Not only is formation error low, but it



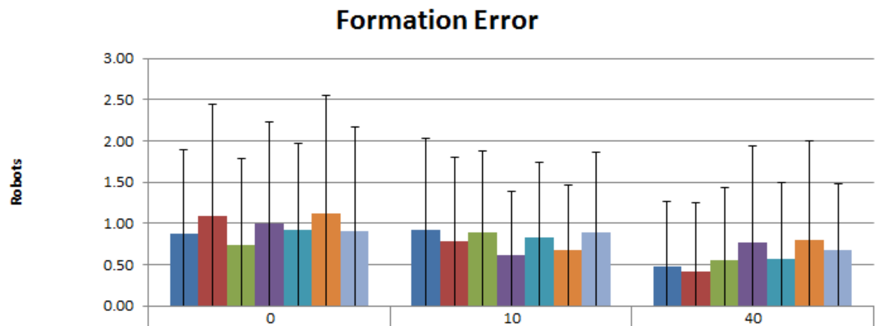
	0	10	40
Sensing Profile 0	3.33	3.67	3.27
Sensing Profile 1	3.98	3.98	3.45
Sensing Profile 2	4.02	3.69	3.47
Sensing Profile 3	3.32	4.16	3.47
Sensing Profile 4	4.08	3.47	2.86
Sensing Profile 5	3.49	3.63	3.16
Sensing Profile 6	3.30	3.39	3.22

Chance of Failure (%)

Figure 4.16: The relationship between the chance of failure, and the number of robots in the largest formation at the end of the trial.

appears unaffected by any of the variations in sensing. There are no clear trends in formation error as the chance of failure increases. One explanation for the higher error rates is the fact that most of the trials did not result in all robots in a single formation, and were therefore cut off at an arbitrary point.

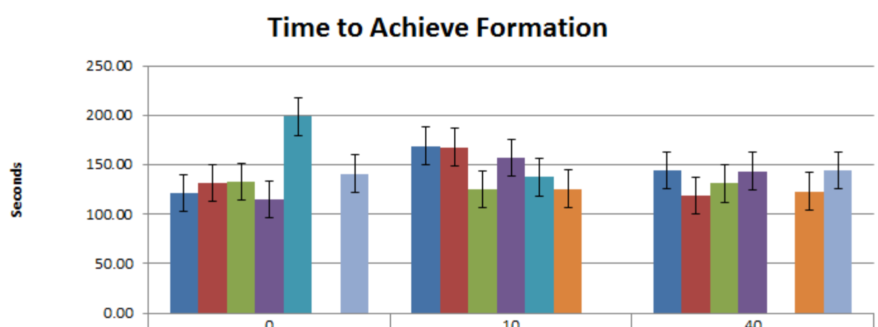
As mentioned in Section 4.5.1, the large error bars on the end of trial metrics are due to trials that were interrupted in the middle of a balancing or merging operation.



	0	10	40
Sensing Profile 0	0.88	0.92	0.48
Sensing Profile 1	1.08	0.78	0.41
Sensing Profile 2	0.74	0.88	0.55
Sensing Profile 3	1.00	0.61	0.76
Sensing Profile 4	0.92	0.82	0.57
Sensing Profile 5	1.12	0.67	0.80
Sensing Profile 6	0.90	0.88	0.67

Chance of Failure (%)

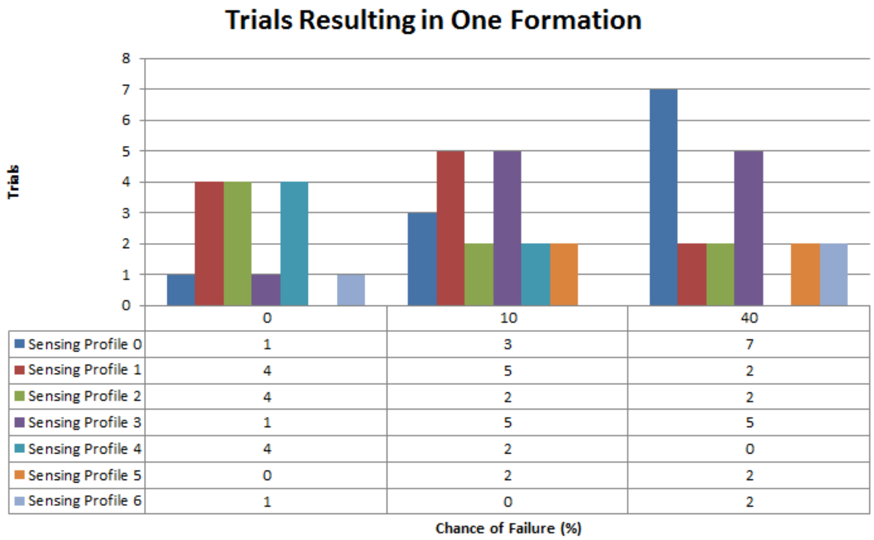
Figure 4.17: The relationship between the chance of failure, and the formation error at the end of the trial.



	0	10	40
Sensing Profile 0	121.00	169.00	144.14
Sensing Profile 1	131.25	167.60	119.00
Sensing Profile 2	132.75	125.00	131.00
Sensing Profile 3	115.00	156.80	143.40
Sensing Profile 4	198.75	137.50	
Sensing Profile 5		125.50	123.00
Sensing Profile 6	141.00		144.50

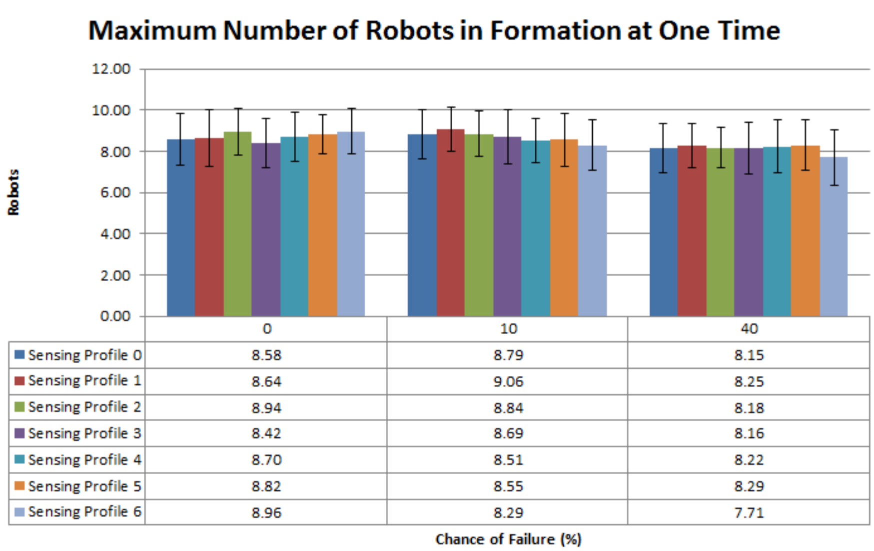
Chance of Failure (%)

Figure 4.18: The relationship between the chance of robot failure and the amount of time it takes to establish a single formation.



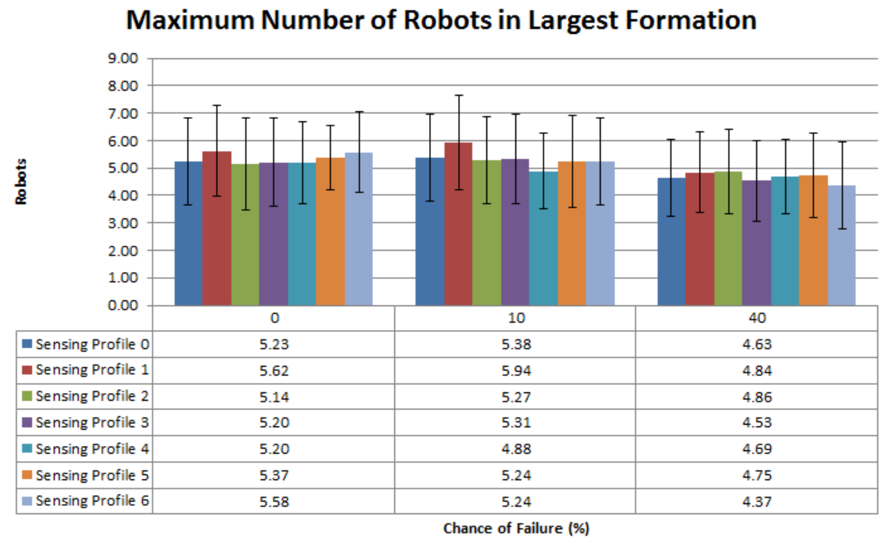
	0	10	40
Sensing Profile 0	1	3	7
Sensing Profile 1	4	5	2
Sensing Profile 2	4	2	2
Sensing Profile 3	1	5	5
Sensing Profile 4	4	2	0
Sensing Profile 5	0	2	2
Sensing Profile 6	1	0	2

Figure 4.19: The number of trials resulting in a single formation at the end as chance of robot failure is increased.



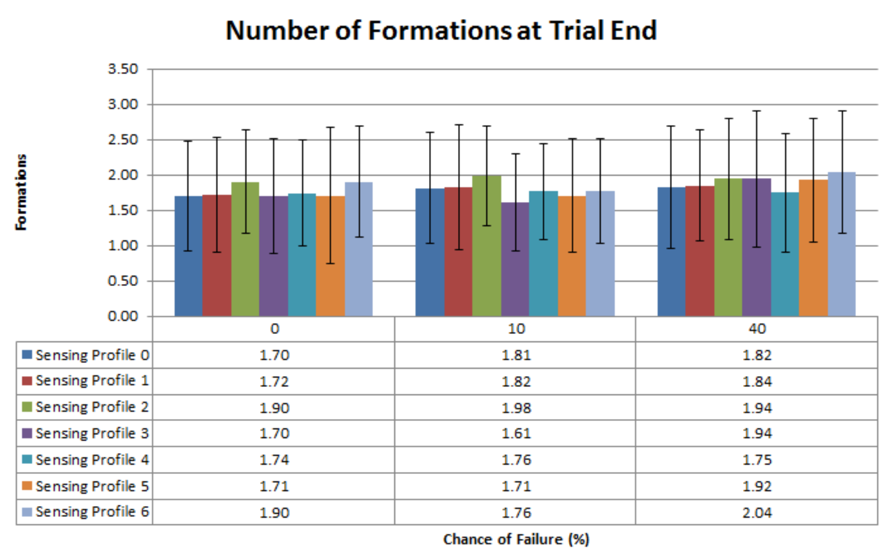
	0	10	40
Sensing Profile 0	8.58	8.79	8.15
Sensing Profile 1	8.64	9.06	8.25
Sensing Profile 2	8.94	8.84	8.18
Sensing Profile 3	8.42	8.69	8.16
Sensing Profile 4	8.70	8.51	8.22
Sensing Profile 5	8.82	8.55	8.29
Sensing Profile 6	8.96	8.29	7.71

Figure 4.20: The maximum number of robots to be participating in formations at the same time as chance of robot failure is increased.



	0	10	40
Sensing Profile 0	5.23	5.38	4.63
Sensing Profile 1	5.62	5.94	4.84
Sensing Profile 2	5.14	5.27	4.86
Sensing Profile 3	5.20	5.31	4.53
Sensing Profile 4	5.20	4.88	4.69
Sensing Profile 5	5.37	5.24	4.75
Sensing Profile 6	5.58	5.24	4.37

Figure 4.21: The maximum number of robots in the largest formation as chance of robot failure increases.



	0	10	40
Sensing Profile 0	1.70	1.81	1.82
Sensing Profile 1	1.72	1.82	1.84
Sensing Profile 2	1.90	1.98	1.94
Sensing Profile 3	1.70	1.61	1.94
Sensing Profile 4	1.74	1.76	1.75
Sensing Profile 5	1.71	1.71	1.92
Sensing Profile 6	1.90	1.76	2.04

Figure 4.22: The number of distinct formations at the end of the trial, as chance of robot failure increases.

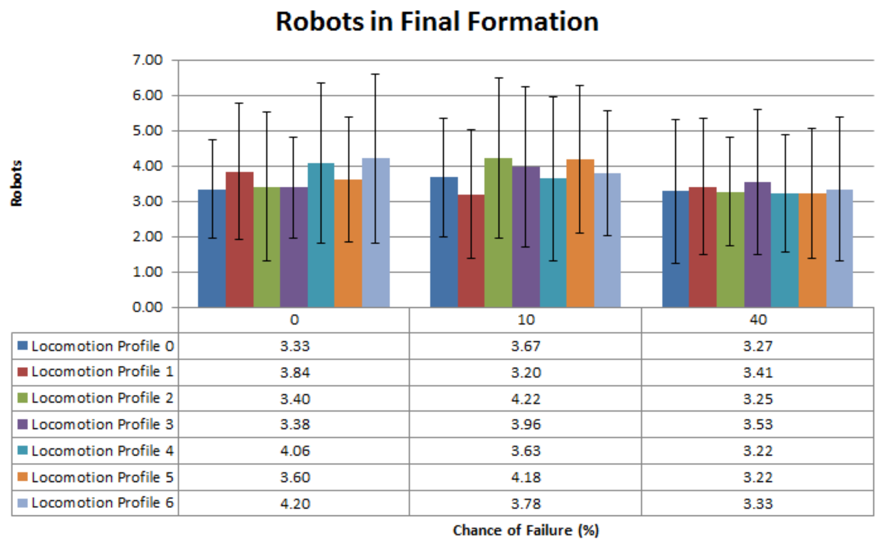
Varying Locomotion

The results of varying locomotion as chance to failure was increased were similar to the results of varying sensing.

Figure 4.23 shows that in a 10 robot trial, between 3 and 4 robots were in the largest formation at the end of a trial. This does not seem very good, until it is combined with the results in Figure 4.26 it is clear that most of the trials in question did not end with a proper formation. This means that they were cut off at some arbitrary point. They may have been in the middle of a merging or balancing operation. Two better measurements to use in such a case are the results in figures 4.27 and 4.28. These charts clearly show that over the course of a trial, on average 8 or more of the 10 robots in a trial participate in a formation. Of those, 5 to 6 participated in the largest formation. This indicates that we likely have several distinct formations forming throughout the trials. This case is illustrated in Figure 4.1 and described in section 4.2.

The time to achieve formation, described by Figure 4.25 seems to suggest that robot failure chance has little effect on the time to establish a formation, nor does the level of robot locomotion. When looking more closely, and examining the data in Figure 4.26, it becomes clear that there is too little data to conclusively say this.

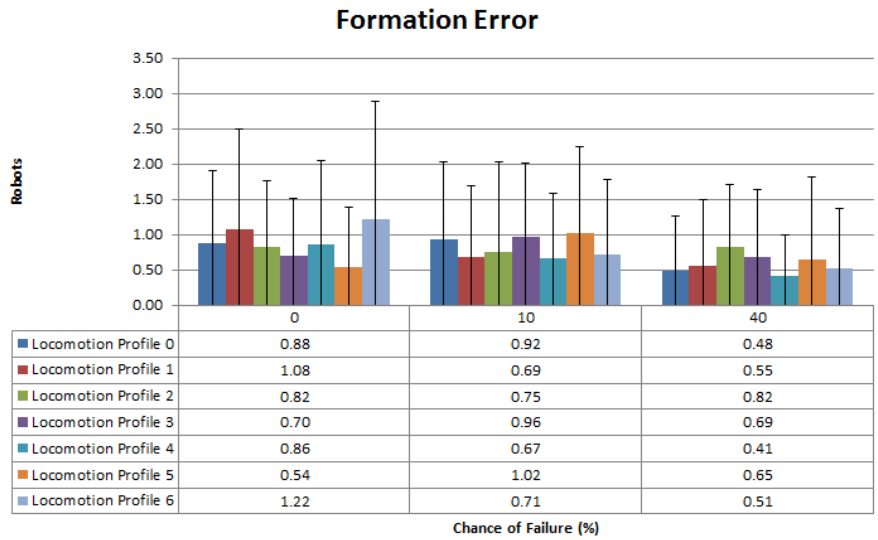
Figure 4.24 shows low formation error, it also appears that formation error is unaffected by any of the variations in locomotion. There are no clear trends in formation error as the chance of failure increases. One explanation for the higher error rates is the fact that most of the trials did not result in all robots in a single formation, and were therefore cut off at an arbitrary point.



	0	10	40
Locomotion Profile 0	3.33	3.67	3.27
Locomotion Profile 1	3.84	3.20	3.41
Locomotion Profile 2	3.40	4.22	3.25
Locomotion Profile 3	3.38	3.96	3.53
Locomotion Profile 4	4.06	3.63	3.22
Locomotion Profile 5	3.60	4.18	3.22
Locomotion Profile 6	4.20	3.78	3.33

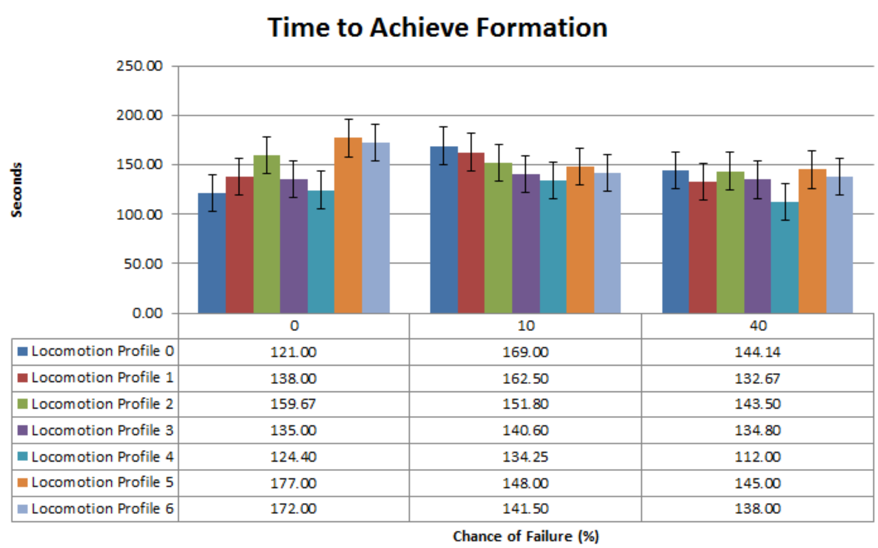
Figure 4.23: The relationship between the chance of failure, and the number of robots in the largest formation at the end of the trial.

As mentioned in Section 4.5.1, the large error bars on the end of trial metrics are due to trials that were interrupted in the middle of a balancing or merging operation.



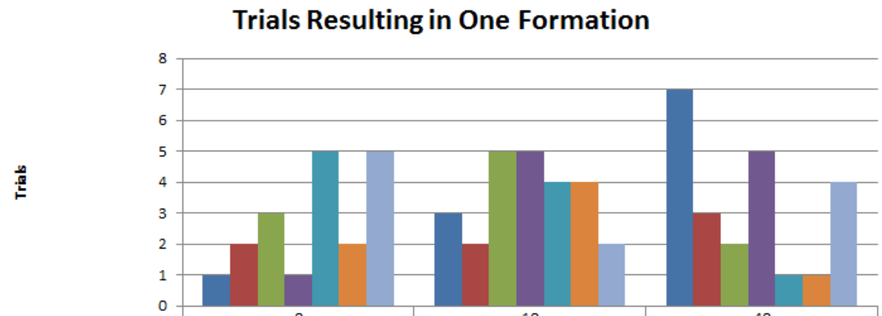
	0	10	40
Locomotion Profile 0	0.88	0.92	0.48
Locomotion Profile 1	1.08	0.69	0.55
Locomotion Profile 2	0.82	0.75	0.82
Locomotion Profile 3	0.70	0.96	0.69
Locomotion Profile 4	0.86	0.67	0.41
Locomotion Profile 5	0.54	1.02	0.65
Locomotion Profile 6	1.22	0.71	0.51

Figure 4.24: The relationship between the chance of failure, and the formation error at the end of the trial.



	0	10	40
Locomotion Profile 0	121.00	169.00	144.14
Locomotion Profile 1	138.00	162.50	132.67
Locomotion Profile 2	159.67	151.80	143.50
Locomotion Profile 3	135.00	140.60	134.80
Locomotion Profile 4	124.40	134.25	112.00
Locomotion Profile 5	177.00	148.00	145.00
Locomotion Profile 6	172.00	141.50	138.00

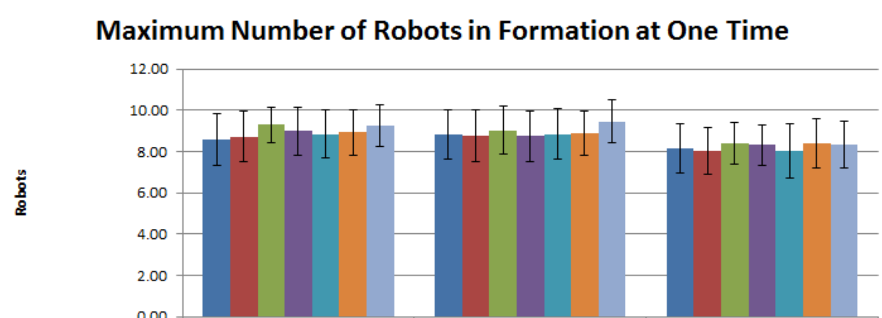
Figure 4.25: The relationship between the chance of robot failure and the amount of time it takes to establish a single formation.



	0	10	40
Locomotion Profile 0	1	3	7
Locomotion Profile 1	2	2	3
Locomotion Profile 2	3	5	2
Locomotion Profile 3	1	5	5
Locomotion Profile 4	5	4	1
Locomotion Profile 5	2	4	1
Locomotion Profile 6	5	2	4

Chance of Failure (%)

Figure 4.26: The number of trials resulting in a single formation at the end as chance of robot failure is increased.



	0	10	40
Locomotion Profile 0	8.58	8.79	8.15
Locomotion Profile 1	8.72	8.75	8.04
Locomotion Profile 2	9.30	9.02	8.41
Locomotion Profile 3	8.98	8.75	8.31
Locomotion Profile 4	8.84	8.84	8.04
Locomotion Profile 5	8.92	8.88	8.37
Locomotion Profile 6	9.24	9.45	8.31

Chance of Failure (%)

Figure 4.27: The maximum number of robots to be participating in formations at the same time as chance of robot failure is increased.

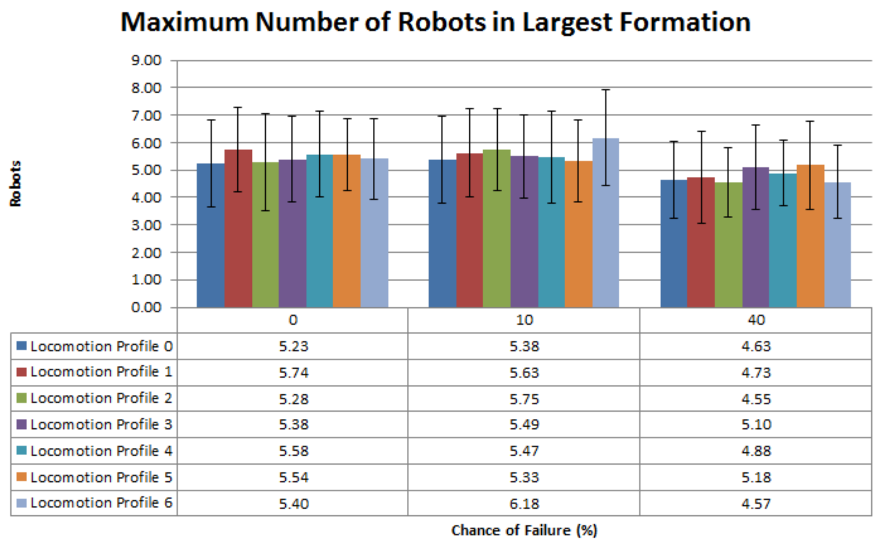


Figure 4.28: The maximum number of robots in the largest formation as chance of robot failure increases.

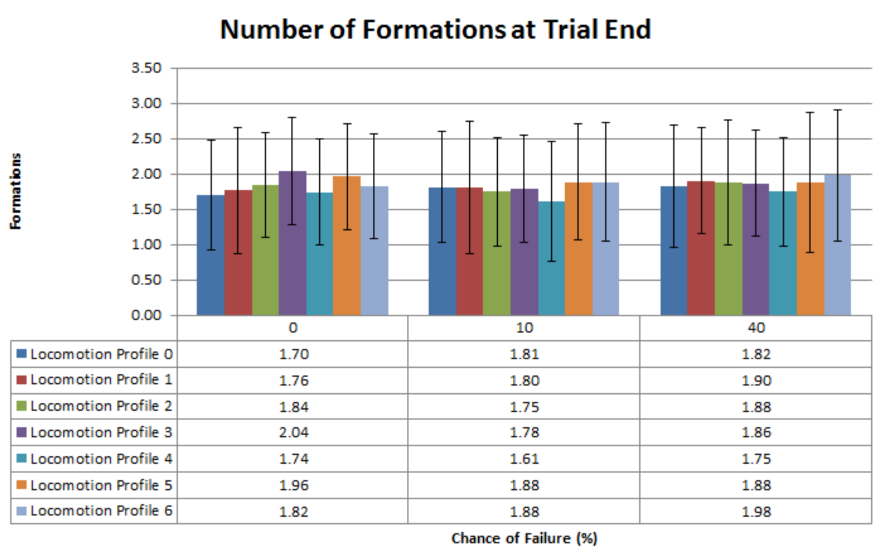


Figure 4.29: The number of distinct formations at the end of the trial, as chance of robot failure increases.

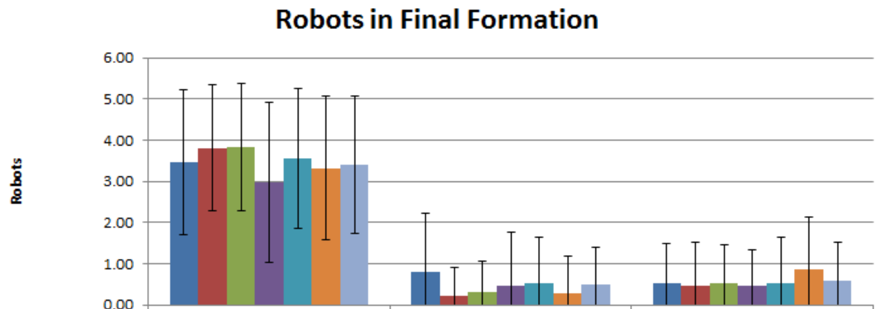
4.5.4 Adaptability Trials

Adaptability trials were carried out using five robots, with no chance of robot failure.

Adaptability results were not as strong as expected. One positive aspect of these trials is that they were consistent as the sensing and locomotion capabilities of the robots were varied.

Varying Sensing

The effect of varying sensing on the reported error metrics is small for all adaptability trials. There is a clear decrease in the number of trials that result in all robots in a single formation as we move to the moderate and hard worlds. This relationship is evident in Figure 4.33. This means that the results presented in figures 4.32, 4.36 and 4.30 are less relevant to the discussion. All of these metrics show poor performance, indicating that the method is not very adaptable. these metrics are all susceptible to the fact that they are sampled only at the end of the trial, and the formation in question may be in the process of merging or balancing at the time. Figure 4.36, however, shows very low numbers of distinct formations at the end of the trial. This indicates is a likely indicator that large numbers of robots are being separated by obstacles. More useful statistics come from Figure 4.34 This graph indicates that there are points in time when 3 to 4 robots are participating in a formation, even in the most difficult world. Figure 4.35 shows a slight drop in the maximum number of robots in the largest formation, as we move to the more challenging worlds. In the easy world, we see between 2 and 3 robots in the largest formation on average. When



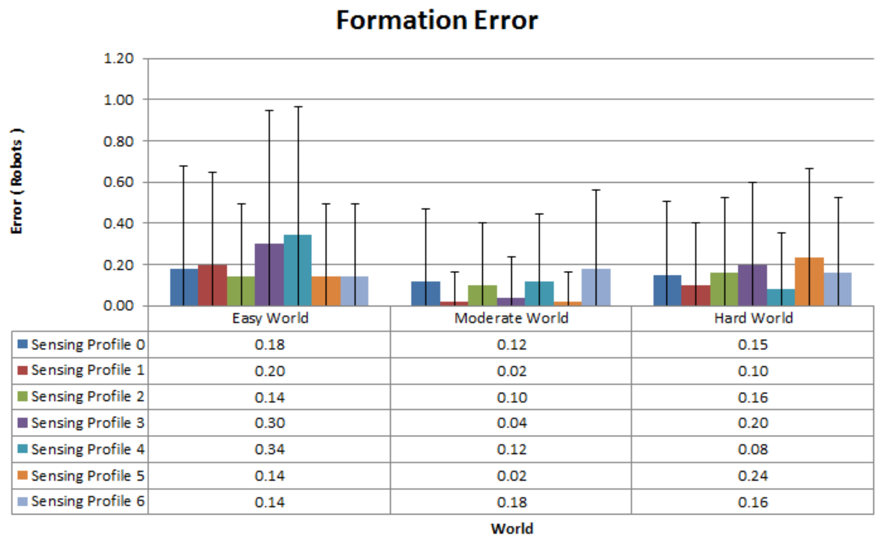
	Easy World	Moderate World	Hard World
Sensing Profile 0	3.46	0.79	0.53
Sensing Profile 1	3.80	0.22	0.45
Sensing Profile 2	3.84	0.29	0.51
Sensing Profile 3	2.96	0.47	0.47
Sensing Profile 4	3.56	0.53	0.53
Sensing Profile 5	3.32	0.27	0.84
Sensing Profile 6	3.40	0.49	0.57

World

Figure 4.30: The relationship between the world in use, and the number of robots in the largest formation at the end of the trial.

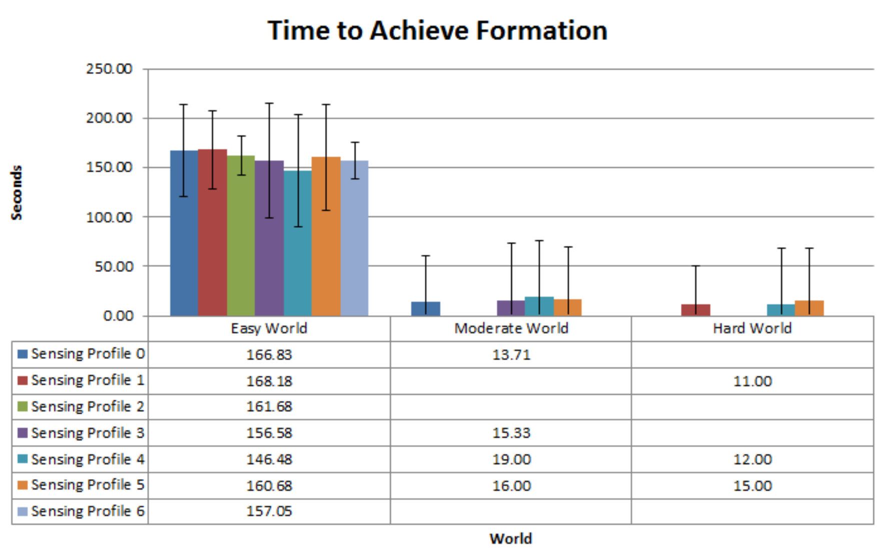
moving to the obstacle-filled worlds, this number drops to around 2. Finally, there is little change in formation error when moving to these more complex worlds. This is illustrated in Figure 4.31 In all cases, the formation error is quite low.

As mentioned in Section 4.5.1, the large error bars on the end of trial metrics are due to trials that were interrupted in the middle of a balancing or merging operation.



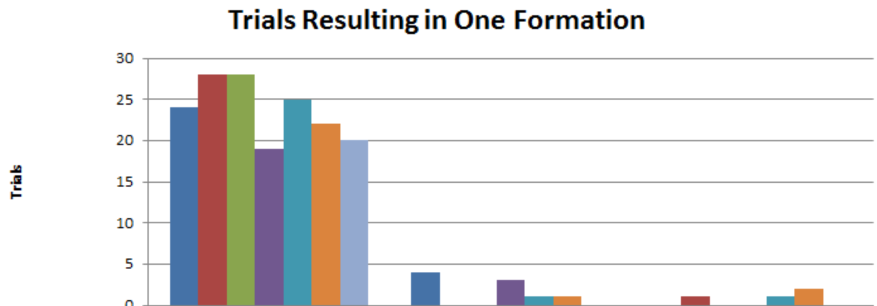
	Easy World	Moderate World	Hard World
Sensing Profile 0	0.18	0.12	0.15
Sensing Profile 1	0.20	0.02	0.10
Sensing Profile 2	0.14	0.10	0.16
Sensing Profile 3	0.30	0.04	0.20
Sensing Profile 4	0.34	0.12	0.08
Sensing Profile 5	0.14	0.02	0.24
Sensing Profile 6	0.14	0.18	0.16

Figure 4.31: The relationship between the world in use, and the formation error at the end of the trial.



	Easy World	Moderate World	Hard World
Sensing Profile 0	166.83	13.71	
Sensing Profile 1	168.18		11.00
Sensing Profile 2	161.68		
Sensing Profile 3	156.58	15.33	
Sensing Profile 4	146.48	19.00	12.00
Sensing Profile 5	160.68	16.00	15.00
Sensing Profile 6	157.05		

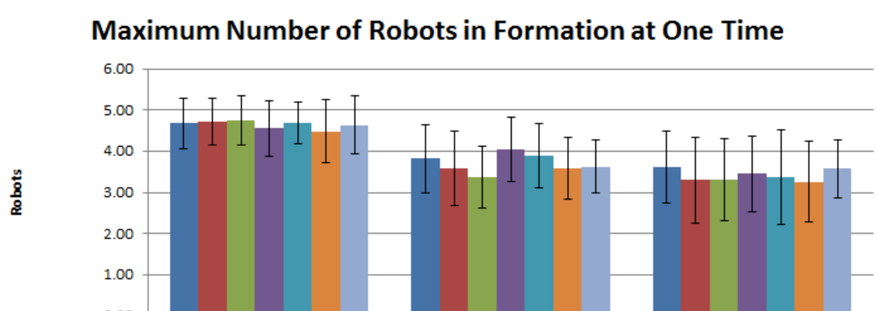
Figure 4.32: The relationship between the world in use and the amount of time it takes to establish a single formation.



	Easy World	Moderate World	Hard World
Sensing Profile 0	24	4	0
Sensing Profile 1	28	0	1
Sensing Profile 2	28	0	0
Sensing Profile 3	19	3	0
Sensing Profile 4	25	1	1
Sensing Profile 5	22	1	2
Sensing Profile 6	20	0	0

World

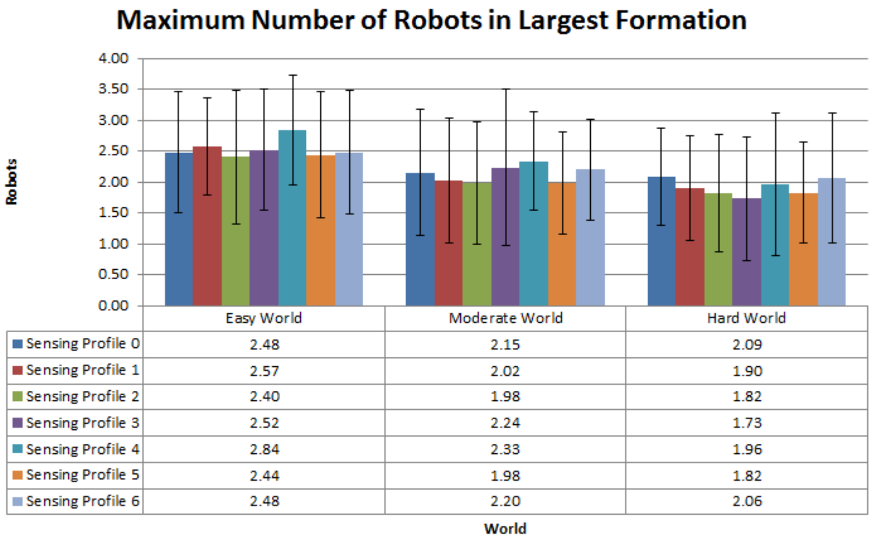
Figure 4.33: The number of trials resulting in a single formation at the end as the world in use changes.



	Easy World	Moderate World	Hard World
Sensing Profile 0	4.67	3.81	3.61
Sensing Profile 1	4.73	3.59	3.29
Sensing Profile 2	4.74	3.37	3.31
Sensing Profile 3	4.56	4.04	3.45
Sensing Profile 4	4.70	3.90	3.37
Sensing Profile 5	4.48	3.59	3.25
Sensing Profile 6	4.64	3.63	3.57

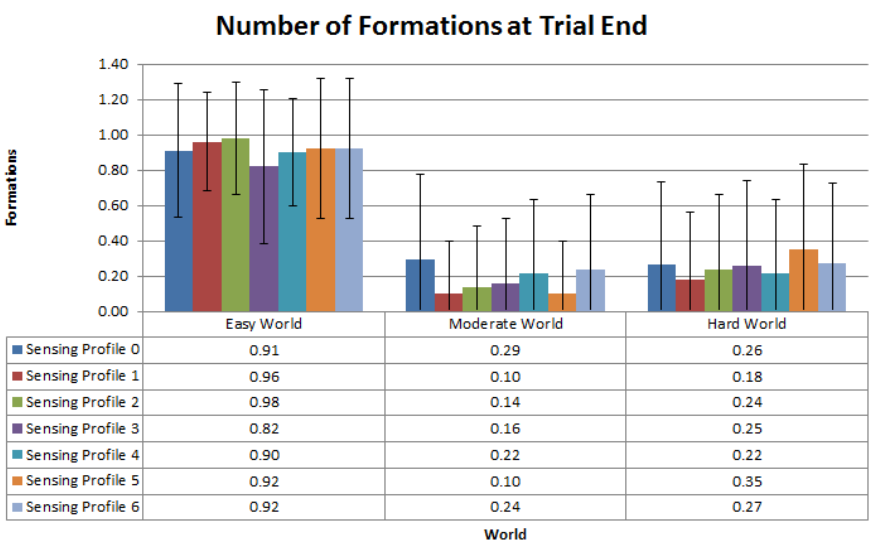
World

Figure 4.34: The maximum number of robots to be participating in formations at the same time as the world in use changes.



	Easy World	Moderate World	Hard World
Sensing Profile 0	2.48	2.15	2.09
Sensing Profile 1	2.57	2.02	1.90
Sensing Profile 2	2.40	1.98	1.82
Sensing Profile 3	2.52	2.24	1.73
Sensing Profile 4	2.84	2.33	1.96
Sensing Profile 5	2.44	1.98	1.82
Sensing Profile 6	2.48	2.20	2.06

Figure 4.35: The maximum number of robots in the largest formation as the world in use changes.



	Easy World	Moderate World	Hard World
Sensing Profile 0	0.91	0.29	0.26
Sensing Profile 1	0.96	0.10	0.18
Sensing Profile 2	0.98	0.14	0.24
Sensing Profile 3	0.82	0.16	0.25
Sensing Profile 4	0.90	0.22	0.22
Sensing Profile 5	0.92	0.10	0.35
Sensing Profile 6	0.92	0.24	0.27

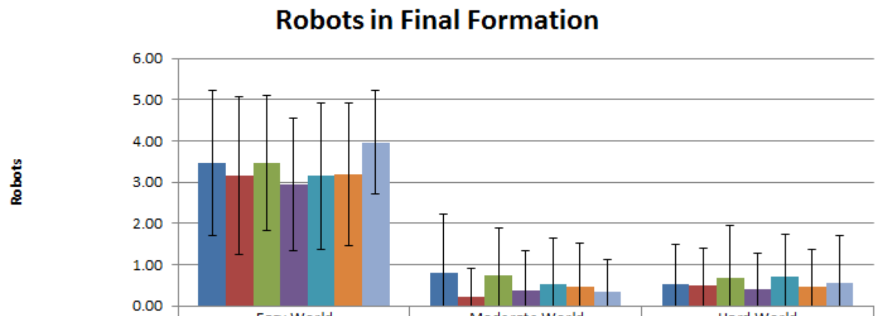
Figure 4.36: The number of distinct formations at the end of the trial, as the world in use changes.

Varying Locomotion

Variations in locomotion profile seem to have little effect on any of the collected metrics in the adaptability trials. Trials involving the medium and hard worlds show very few successful formations. This is evident in Figure 4.40. This means that the results presented in figures 4.39, 4.43 and 4.37 are less relevant to the discussion. Because of the potential that merging or balancing could be in progress at the end of a trial, these metrics are less meaningful than the others. Figure 4.36, however, shows very low numbers of distinct formations at the end of the average trial. This is a likely indicator that large numbers of robots are being separated from the formation. More useful statistics come from Figure 4.41 This graph indicates that there are points in time when 3 to 4 robots are participating in a formation, even in the most difficult world. Figure 4.42 shows a slight drop in the maximum number of robots in the largest formation, as we move to the more challenging worlds. In the easy world, we see between 2 and 3 robots in the largest formation on average. When moving to the obstacle-filled worlds, this number drops to around 2. Finally, there is little change in formation error when moving to these more complex worlds. This is illustrated in Figure 4.38 In all cases, the formation error is quite low, indicating that balancing is working well.

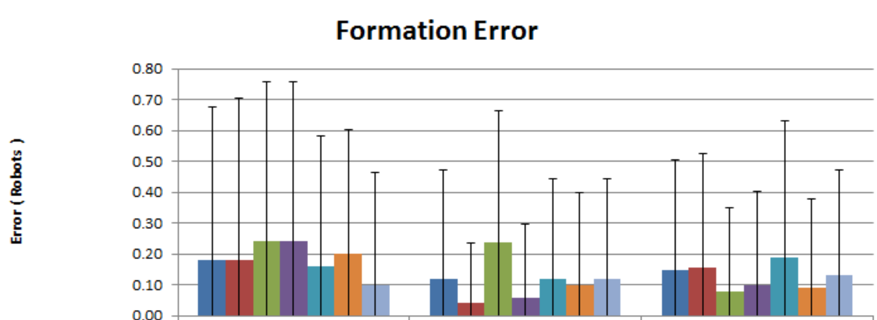
As mentioned in Section 4.5.1, the large error bars on the end of trial metrics are due to trials that were interrupted in the middle of a balancing or merging operation.

Overall, adaptability results seem to suggest that the obstacles are splitting the robots up, and causing them to form smaller groups, or be separated from the group.



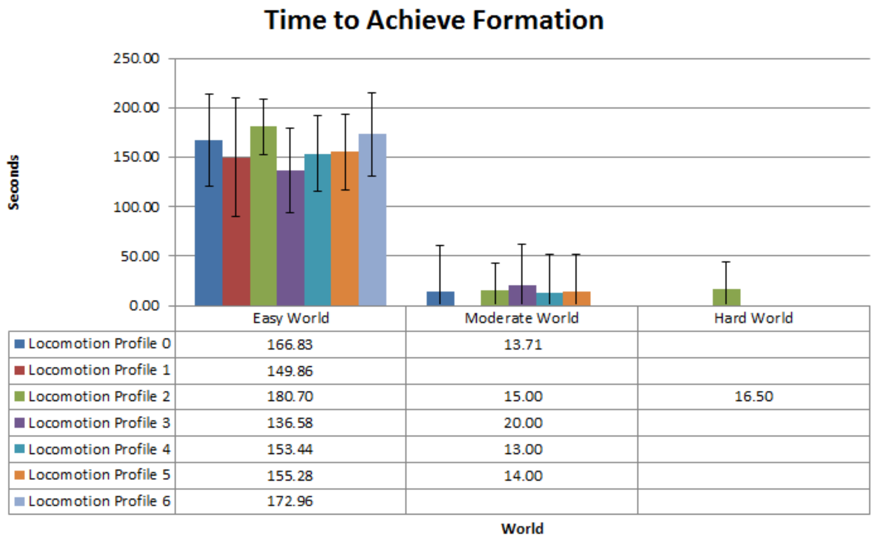
	Easy World	Moderate World	Hard World
Locomotion Profile 0	3.46	0.79	0.53
Locomotion Profile 1	3.16	0.22	0.49
Locomotion Profile 2	3.46	0.75	0.67
Locomotion Profile 3	2.94	0.35	0.40
Locomotion Profile 4	3.14	0.53	0.71
Locomotion Profile 5	3.18	0.47	0.47
Locomotion Profile 6	3.96	0.33	0.55

Figure 4.37: The relationship between the world in use, and the number of robots in the largest formation at the end of the trial.



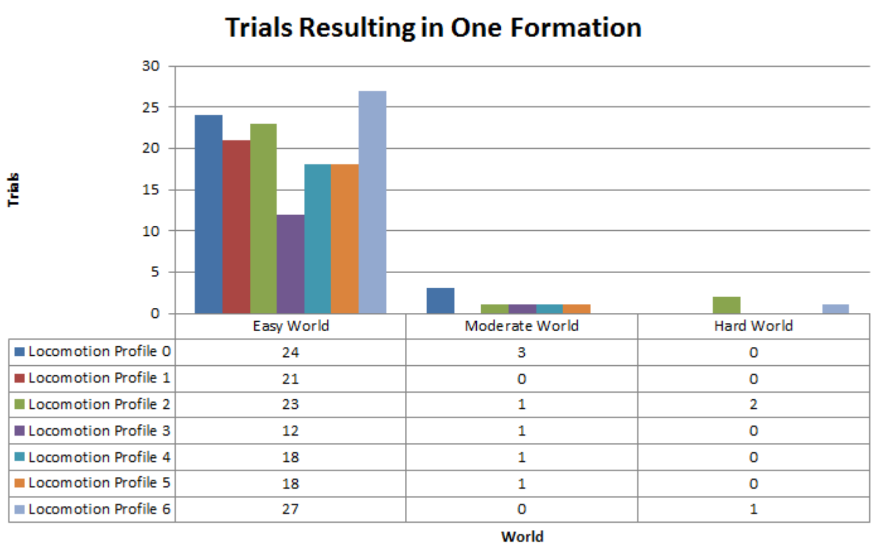
	Easy World	Moderate World	Hard World
Locomotion Profile 0	0.18	0.12	0.15
Locomotion Profile 1	0.18	0.04	0.16
Locomotion Profile 2	0.24	0.24	0.08
Locomotion Profile 3	0.24	0.06	0.10
Locomotion Profile 4	0.16	0.12	0.19
Locomotion Profile 5	0.20	0.10	0.09
Locomotion Profile 6	0.10	0.12	0.13

Figure 4.38: The relationship between the world in use, and the formation error at the end of the trial.



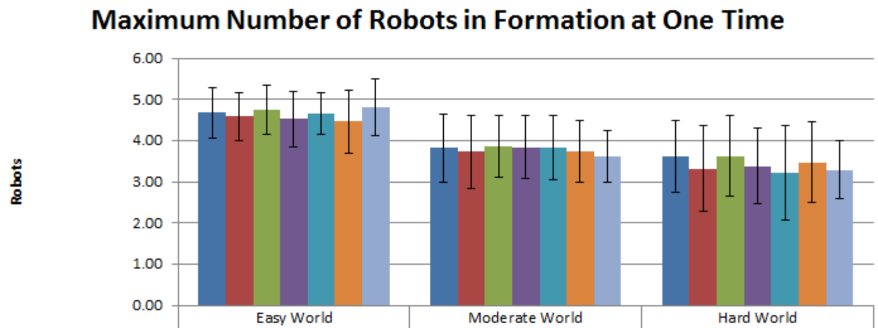
	Easy World	Moderate World	Hard World
Locomotion Profile 0	166.83	13.71	
Locomotion Profile 1	149.86		
Locomotion Profile 2	180.70	15.00	16.50
Locomotion Profile 3	136.58	20.00	
Locomotion Profile 4	153.44	13.00	
Locomotion Profile 5	155.28	14.00	
Locomotion Profile 6	172.96		

Figure 4.39: The relationship between the world in use and the amount of time it takes to establish a single formation.



	Easy World	Moderate World	Hard World
Locomotion Profile 0	24	3	0
Locomotion Profile 1	21	0	0
Locomotion Profile 2	23	1	2
Locomotion Profile 3	12	1	0
Locomotion Profile 4	18	1	0
Locomotion Profile 5	18	1	0
Locomotion Profile 6	27	0	1

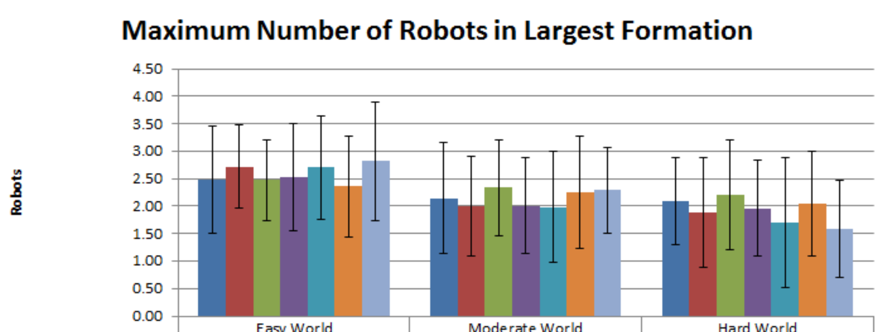
Figure 4.40: The number of trials resulting in a single formation at the end as the world in use changes.



	Easy World	Moderate World	Hard World
Locomotion Profile 0	4.67	3.81	3.61
Locomotion Profile 1	4.58	3.73	3.31
Locomotion Profile 2	4.74	3.86	3.63
Locomotion Profile 3	4.52	3.84	3.38
Locomotion Profile 4	4.66	3.84	3.23
Locomotion Profile 5	4.46	3.75	3.47
Locomotion Profile 6	4.80	3.61	3.29

World

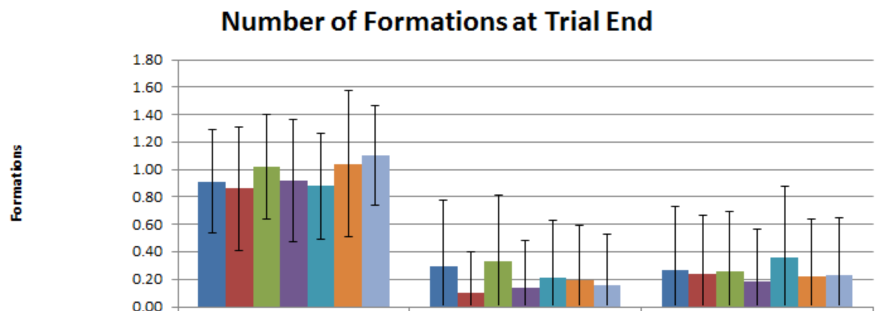
Figure 4.41: The maximum number of robots to be participating in formations at the same time as the world in use changes.



	Easy World	Moderate World	Hard World
Locomotion Profile 0	2.48	2.15	2.09
Locomotion Profile 1	2.72	2.00	1.88
Locomotion Profile 2	2.48	2.33	2.20
Locomotion Profile 3	2.52	2.00	1.96
Locomotion Profile 4	2.70	1.98	1.71
Locomotion Profile 5	2.36	2.25	2.04
Locomotion Profile 6	2.82	2.29	1.58

World

Figure 4.42: The maximum number of robots in the largest formation as the world in use changes.



	Easy World	Moderate World	Hard World
Locomotion Profile 0	0.91	0.29	0.26
Locomotion Profile 1	0.86	0.10	0.24
Locomotion Profile 2	1.02	0.33	0.25
Locomotion Profile 3	0.92	0.14	0.18
Locomotion Profile 4	0.88	0.22	0.35
Locomotion Profile 5	1.04	0.20	0.22
Locomotion Profile 6	1.10	0.16	0.23

World

Figure 4.43: The number of distinct formations at the end of the trial, as the world in use changes.

4.6 Summary

In this chapter, I have described the methods that I used to evaluate my approach. I have established metrics to be used for comparison, and described my experimental design, and the results of my trials. I established that my approach to formation control is scalable to larger numbers of robots. I also established that it is robust against failure of any one robot.

Chapter 5

Conclusion

5.1 Introduction

The chapter that follows summarizes the contributions of my thesis, and addresses the answers to my research questions. It also examines potential future work that I would like to undertake on this topic.

5.2 Contributions

The major contribution of my research is a formation control technique capable of achieving and maintaining formation in a real-world environment. This technique allows for dynamic variability in the number of robots participating in the formation. Through this technique, I have demonstrated that heterogeneous robots can assemble into and maintain formations, using only local sensing and communication.

I demonstrated the impact of robot heterogeneity on robustness, scalability and adaptability of my formation control technique. By extension, I have shown that a formation control technique can be robust, adaptable and scalable with a very small amount of global data and without greatly restricting the number of possible formations.

5.3 Answers to Research Questions

The results presented in chapter 4 can be used to address my research questions, which were presented in chapter 1.

Can the scalability of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? The numbers in chapter 4 demonstrate that this technique is indeed scalable to larger number of robots. As the number of robots grows, so does the maximum number of robots in formation. This indicates that although some robots fall away, and are not able to participate in the formation, a large number of them do. The low formation error numbers show that the balancing algorithm works properly, even with larger numbers of robots. As the sensing and locomotion capabilities of the robots are varied, there is little to no impact on the aggregate results. Formation error remains low, while the number of robots in formation remains high.

In the scalability trials, I discovered that although my approach is scalable to large numbers of robots. The current implementation often does not produce a single formation. Whether or not this is acceptable is largely domain specific. In domains where a large number of comparatively small formations is a positive result, my technique is quite valuable.

Can the robustness of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? The results from chapter 4 demonstrate that this is a robust technique. As robots fail, the approach is clearly able to create and maintain correct formations. As with the scalability trials, the results imply that the approach generally results in multiple small formations, but most robots do participate in formation.

As with the scalability results, the utility of this approach depends on whether or not the application can tolerate a larger number of small formations, or if it requires a single formation. In the former case, my approach is a good choice.

Can the adaptability of a formation control approach be preserved, in a decentralized approach, even though robots have differing levels of ability in locomotion control and perception? Although the base approach was not as adaptable to new environments as I would have liked, robot heterogeneity did not seem to have any detrimental impact on the adaptability of the formation control approach.

In general, the metrics shown in section 4.5 are quite conservative. If a single formation is not achieved, the end of trial metrics can be quite harsh on an otherwise successful trial. This conservative nature is useful when evaluating an approach designed to yield a single formation, but it unfairly penalizes an approach that successfully achieves a group of smaller formations. The reason that these metrics are conservative is a combination of the metrics themselves, and the trial end conditions. I discuss an alternative trial end condition in section 5.4. The new end condition could help to make the end of trial metrics more meaningful.

Overall, my approach handled robot heterogeneity very well. There were no notable differences between trials involving any of the different sensing or locomotion profiles. This demonstrates that my approach handles robot heterogeneity well.

5.4 Future Work

As future work, I would like to investigate further degrees of robot heterogeneity in formation control. It would be interesting to examine how robots with very different physical characteristics (e.g. a combination of humanoid and wheeled robots, as in Kiener and von Stryk [2007]), could establish and maintain formations.

Another interesting extension to my work would be physical trials involving my approach. These trials would help to further validate the effectiveness of my approach, and would prove that it is ready for realistic environments. Some of the simplifying assumptions that I have made in my simulation work would have to be abandoned in a physical domain. The most notable such assumption is the selection of a pre-defined destination for all robots. In a physical domain, robots would need some sort of system or marker to ensure that they were heading in the right direction.

In addition to working with physical robots, it would be interesting to evaluate my approach in a mixed reality environment Anderson et al. [2009]. This would allow for greater complexity in the environment without the need to build elaborate testing facilities. It also allows for a less contrived way of examining robot failure. The mixed-reality environment could simulate fire, or holes in the terrain. Instead of having a robot fail randomly, they could fail as a result of environmental conditions. This is much more realistic.

I would also like to implement an approach to keeping robots together. In all of my trials, robots started close together, but in many of the larger trials, it became clear that robots were splitting off into groups, and creating smaller formations. Anecdotally, I have noticed this happening frequently when robots are waiting for their target to pass them. They slow down, and sometimes the formation moves just out of their visual range. It would be interesting to explore methods of making sure that robots do not move out of perceptual range. There are several approaches that I can think of for this. The first would be a *slow down* message. This message would instruct a robot to reduce its speed, and it could be sent as a robot moves close to a robot's maximum sensing range. Another approach would be to have robots that don't perceive any others periodically perform some sort of search pattern. (e.g. the spiraling method described in Ghoshal and Shell [2011])

Another interesting extension to my approach would be a variable obstacle height. This could better simulate the sensing challenges faced by robots in real world domains. Obstacles may be passable to some formation members, but not all. This would create an interesting challenge, and would likely require some way of specifying the maximum obstacle height that a formation as a whole could pass over. This could be handled in the same way as maximum speed.

One interesting piece of future work would be to develop a new stopping condition for my trials. The current stopping condition is arbitrary, and penalizes formations that are actively merging or balancing at the trial end time. It would be interesting to extend the trial by a number of seconds, each time a join or balance operation occurs. This could help to give more reliable results.

5.5 Conclusion

In this thesis, I have examined how robot heterogeneity affects the ability of robots to establish and maintain formations. To this end, I have developed a new approach to formation control. Although my approach does not guarantee successful formation creation, it does so in the vast majority of cases. Through my work, I have shown that formations consisting of heterogeneous robots can establish and maintain formations in a way that is scalable, robust and adaptable.

Bibliography

J. Anderson, J. Baltes, and K.-Y. Tu. Improving robotics competitions for real-world evaluation of AI. In *Proceedings of the AAAI Spring Symposium on Experimental Design for Real-World Systems*, pages 1–8, Stanford, CA, March 2009.

R. C. Arkin. *Behavior-based Robotics*. MIT Press, Cambridge, 1998.

T. Balch and R. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14:926–939, December 1998.

A. Billard and K. Dautenhahn. Experiments in learning by imitation - grounding and use of communication in robotic agents. *Adaptive Behavior*, 7(3/4):415–438, 1999.

A. Brandao, F. Martins, V. Rampinelli, M. Sarcinelli-Filho, T. Bastos-Filho, and R. Carelli. A multi-layer control scheme for multi-robot formations with adaptive dynamic compensation. In *ICM 2009. IEEE International Conference on Mechatronics*, pages 1–6, April 2009. doi: 10.1109/ICMECH.2009.4957216.

S. Burion. Human detection for robotic urban search and rescue. Technical report, Swiss Federal Institute of Technology, 2004.

L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques. Stabilization of a hierarchical formation of unicycle robots with velocity and curvature constraints. *IEEE Transactions on Robotics*, 25(5):1176–1184, Oct. 2009. ISSN 1552-3098. doi: 10.1109/TRO.2009.2026505.

A. D'Angelo and E. Pagello. Making collective behaviours to work through implicit communication. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 81–86, april 2005. doi: 10.1109/ROBOT.2005.1570100.

M. de Denus, J. Anderson, and J. Baltes. Heuristic formation control in multi-robot systems using local communication and limited identification. In J. Baltes, M. Lagoudakis, T. Naruse, and S. Shiry, editors, *Robocup-2009: Robot Soccer World Cup XIII*, pages 437–448. Springer, Berlin, 2010.

T. Dierks and S. Jagannathan. Neural network control of mobile robot formations using rise feedback. *IEEE Transactions on Systems, Man, and Cybernetics*, 39(2): 332–347, April 2009. ISSN 1083-4419. doi: 10.1109/TSMCB.2008.2005122.

Y. Dieudonné, O. Labbani-Igbida, and F. Petit. Circle formation of weak mobile robots. *ACM Trans. Auton. Adapt. Syst.*, 3(4):1–20, 2008. ISSN 1556-4665. doi: http://doi-.acm.org.proxy1.lib.umanitoba.ca/10.1145/1452001.1452006.

P. S. Dutta and S. Sen. Forming stable partnerships. *Cognitive Systems Research*, 4 (3):211–221, 2003.

J. Fredslund and M. Mataric. A general algorithm for robot formations using local sensing and minimal communication. *IEEE Transactions on Robotics and Automation*, 18(5):837–846, 2002.

B. Gerkey, R. Vaughan, K. Stoy, A. Howard, G. Sukhatme, and M. Mataric. Most valuable player: a robot device server for distributed control. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 1226–1231, November 2001.

A. Ghoshal and D. A. Shell. Being there, being the rrt: Space-filling and searching in place with minimalist robots. In *2011 AAAI spring symposium on Multirobot Systems and Physical Data Structures*, 2011.

S. Gilbert, N. Lynch, S. Mitra, and T. Nolte. Self-stabilizing robot formations over unreliable networks. *ACM Trans. Auton. Adapt. Syst.*, 4(3):1–29, 2009. ISSN 1556-4665. doi: http://doi.acm.org.proxy1.lib.umanitoba.ca/10.1145/1552297.1552300.

P. Goldman. Flocking as a possible predator defense in dark-eyed juncos. *Wilson Bulletin*, 92:88–95, 1980.

R. Hadaegh, B. Kang, and D. Scharf. Rule-based formation estimation of distributed spacecraft. In *Fuzzy Systems, 2001. The 10th IEEE International Conference on*, volume 3, pages 1331–1336, 2001. doi: 10.1109/FUZZ.2001.1008904.

D. Harabor and A. Botea. Hierarchical path planning for multi-size agents in heterogeneous environments. In *Computational Intelligence and Games, 2008. CIG '08. IEEE Symposium On*, pages 258–265, dec. 2008. doi: 10.1109/CIG.2008.5035648.

G. Hattenberger, S. Lacroix, and R. Alami. Formation flight: Evaluation of autonomous configuration control algorithms. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2628–2633, San Jose, USA, October 2007.

Y. He and J. Han. Multiple robots formation control based on receding horizon optimization. In *ICIA '09. International Conference on Information and Automation*, pages 734–739, June 2009. doi: 10.1109/ICINFA.2009.5205018.

A. Howard, L. E. Parker, and G. S. Sukhatme. Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection. *The International Journal of Robotics Research*, 1(5–6):431–447, June 2006.

J. Kiener and O. von Stryk. Cooperation of heterogeneous, autonomous robots: A case study of humanoid and wheeled robots. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 959–964, 29 2007-nov. 2 2007. doi: 10.1109/IROS.2007.4399291.

C. Kitts and I. Mas. Cluster space specification and control of mobile multirobot systems. *IEEE/ASME Transactions on Mechatronics*, 14(2):207–218, April 2009. ISSN 1083-4435. doi: 10.1109/TMECH.2009.2013943.

M. Kumar and D. Garg. Aggregation of heterogeneous units in a swarm of robotic agents. In *Resilient Control Systems (ISRCS), 2011 4th International Symposium on*, pages 107–112, aug. 2011. doi: 10.1109/ISRCS.2011.6016099.

D. Kurabayashi, T. Choh, J. Cheng, and T. Funato. Adaptive formation transition among a mobile robot group based on phase gradient. In *ROBIO 2008. IEEE International Conference on Robotics and Biomimetics*, pages 2001–2006, Feb. 2009. doi: 10.1109/ROBIO.2009.4913308.

J.-W. Kwon and D. Chwa. Hierarchical formation control based on a vector field method for wheeled mobile robots. *Robotics, IEEE Transactions on*, 28(6):1335–1345, December 2012. ISSN 1552-3098. doi: 10.1109/TRO.2012.2206869.

G. Lee and N. Y. Chong. Adaptive self-configurable robot swarms based on local interactions. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4182–4187, San Jose, USA, October 2007.

F. Legras and C. Tessier. *Advances in Agent Communication*, chapter LOTTO: Group Formation by Overhearing in Large Teams, page 1956. Springer Berlin, 2004.

M. Mataric. Minimizing complexity in controlling a mobile robot population. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, volume 1, pages 830–835, May 1992a. doi: 10.1109/ROBOT.1992.220192.

M. Mataric. Behavior-based systems: Main properties and implications. In *Proceedings, IEEE International Conference on Robotics and Automation, Workshop on Architectures for Intelligent Control Systems*, pages 46–54, May 1992b.

N. Moshtagh, N. Michael, A. Jadbabaie, and K. Daniilidis. Vision-based, distributed control laws for motion coordination of nonholonomic robots. *IEEE Transactions on Robotics*, 25(4):851–860, Aug. 2009. ISSN 1552-3098. doi: 10.1109/TRO.2009.2022439.

V. Rampinelli, A. Brandao, F. Martins, M. Sarcinelli-Filho, and R. Carelli. A multi-layer control scheme for multi-robot formations with obstacle avoidance. In *ICAR 2009. International Conference on Advanced Robotics*, pages 1–6, June 2009.

A. Ray, P. Benavidez, L. Behera, and M. Jamshidi. Decentralized motion coordination for a formation of rovers. *IEEE Systems Journal*, 3(3):369–381, Sept. 2009. ISSN 1932-8184. doi: 10.1109/JSYST.2009.2031012.

A. Rosenfeld, G. A. Kaminka, S. Kraus, and O. Shehory. A study of mechanisms for improving robotic group performance. *Artificial Intelligence*, 172(6–7):633 – 655, 2008. ISSN 0004-3702. doi: DOI:10.1016/j.artint.2007.09.008. URL `http://www.sciencedirect.com/science/article/B6TYF-4PYYTW5-1/2/73f419bd6132d2644eae0970f1c72c0a`.

J. Shin, H. J. Kim, S. Kim, and Y. Yoon. Formation flight control under communication failure. In *RoboComm '07: Proceedings of the 1st international conference on Robot communication and coordination*, pages 1–4, Piscataway, NJ, USA, 2007. IEEE Press. ISBN 978-963-9799-08-0.

A. Viguria and A. Howard. An integrated approach for achieving multirobot task formations. *IEEE/ASME Transactions on Mechatronics*, 14(2):176–186, April 2009. ISSN 1083-4435. doi: 10.1109/TMECH.2009.2014056.

J. Wang. Establish a globally consistent order of discrete events in distributed robotic systems. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 853 –858 vol.3, May 1993. doi: 10.1109/ROBOT.1993.292251.

N. Wiebe and J. Anderson. Local methods for supporting grounded communication in robot teams. In D. Liu, L. Wang, and K. C. Tan, editors, *Design and Control of Intelligent Robotic Systems*, chapter 14, pages 279–301. Springer-Verlag, Heidelberg, 2009.

H. Yamaguchi. Adaptive formation control for distributed autonomous mobile robot groups. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 3, pages 2300–2305, April 1997.

B. Yamauchi. Packbot: A versatile platform for military robotics. In *Proceedings of SPIE 5422*, pages 228–237, 2004.

E. Yoshida, T. Arai, J. Ota, and T. Miki. Effect of grouping in local communication system of multiple mobilerobots. In *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994.*, volume 2, pages 805–815, Munich, Germany, September 1994a.

E. Yoshida, T. Arai, J. Ota, and T. Miki. Effect of grouping in local communication system of multiple mobilerobots. In *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems '94.*, volume 2, pages 805–815, Munich, Germany, September 1994b.