UNIVERSITY OF ALBERTA

Collective Robotic Intelligence: A Control Theory For Robot Populations

by

Claus Ronald Kube

A Thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science.

Department of Computing Science

Edmonton, Alberta

Spring 1992

ISBN 0-315-73168-0

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: **Claus Ronald Kube**

TITLE OF THESIS: **Collective Robotic Intelligence: A Control Theory For Robot Populations**

DEGREE: **Master of Science**

YEAR THIS DEGREE GRANTED: **1992**

Ron Kube

#1803, 11020 - 53 Avenue
Edmonton, Alberta
Canada T6H 0S4

Date: April 14, 1992

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Collective Robotic Intelligence: A Control Theory For Robot Populations** submitted by **Claus Ronald Kube** in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE.**

Hong Zhang

Dr. Hong Zhang

R. Goebel

Dr. Randy Goebel

Peter van Beek

Dr. Peter van Beek

Roger W Toogood

Dr. Roger Toogood

Date: April 7, 1992

To my grandfather, Herbert J. Thomas,
who taught me the secret of climbing mountains:
never look down and *never* give up.

Abstract

Recently, researchers investigating control architectures for autonomous mobile robots have taken a new approach called behaviour-based control, which has resulted in producing robots with simple insect-like intelligence. All efforts to date have concentrated on designing *single* autonomous robots situated and embodied in the real world. Given that man is not about to invent a highly intelligent autonomous robot tomorrow, we conjecture that useful tasks can be accomplished with today's simple behaviour-based control mechanisms provided multiple robots are organized into collections of task achieving populations. This thesis takes the first steps toward developing a control theory and model for populations of behaviour-based autonomous robots capable of achieving collective tasks without centralized coordination or the use of explicit communication. Specifically, our model is based on several examples of collective behaviour taken from the study of Social Insects. We have tested our collective control strategies by designing a robot population simulator called *SimbotCity*. We have also constructed a system of five homogeneous sensor-based physical robots, capable of achieving simple collective tasks, to demonstrate the feasibility of the proposed theory and model.

Acknowledgements

I would like to take this opportunity to express my sincere thanks to those who have helped me throughout my graduate studies. I am indebted to Dr. Hong Zhang, my supervisor, who initiated the ideas on collective robotics. His guidance has allowed me to pursue my ideas unconstrained, creating an environment in which to conduct research that is both challenging and rewarding.

I would like to also thank my friend Rod Johnson for his encouragement, insightful comments and help in building circuits for, and the filming of, the robots. Special thanks to Denis McLaughlin and Douglas Doole for their skills with a soldering iron.

I am also grateful to Dr. Randy Goebel, Dr. Peter van Beek and Dr. Roger Toogood for the time they took to read and comment on this work.

Finally, I would like to thank my loving wife, Carole, for her patience and encouragement as I happily pursue my dreams.

Contents

List of Figures

Tables

Chapter 1

Intelligent Behaviour

1.1 Introduction

Can intelligent behaviour-based mobile robots achieve tasks collectively? What constitutes intelligent behaviour? When discussing intelligent behaviour is it necessary to put it in the context of human intelligence, animal intelligence, insect intelligence or machine intelligence? Are social insects, which function collectively in groups, intelligent? When viewed as a superorganism, can the collective intelligence of social insects be considered to be greater than the sum of the simple individuals that comprise the group? Can this collective behaviour be simulated on a computer? Better still, can we build useful machines that demonstrate collective behaviour? This chapter attempts to explore these questions and propose solutions whose exploration will concern us throughout the remainder of this thesis.

We begin with the next two sections contrasting the difference between Machine and Robotic Intelligence. Next, we examine a natural example of collective behaviour, the social insects, and conjecture whether a new level of intelligence is possible. We call this new intelligence *Collective Robotic Intelligence* and propose our method for its investigation. Section 1.5 outlines our research goals and objectives and section 1.6 provides an overview of the remainder of the thesis.

1.2 Machine Intelligence

What constitutes intelligent behaviour? Answering this question would depend on the context in which you define intelligence. Are we referring to human, animal, insect or machine intelligence? If we view intelligence as the ability to perceive logical relationships and use one's knowledge to solve problems and respond appropriately to novel situations, then its meaning changes for each case mentioned. Webster (1988) defines behaviour as "the way in which something reacts to its environment." We believe the intelligent behaviour of an agent should be judged solely on the externally observed interaction between the agent and the environment and without regard to its internal mechanisms. Behaviour that we might consider intelligent for an insect is not considered intelligent for a human. Why is that?

Part of the reason may be due to the fact that there is an enormous set of unarticulated background assumptions when we discuss anything that is context sensitive. Our experience has already set our expectations when we discuss the biologically based intelligence. We view human intelligence as being greater than animal intelligence, which is greater than insect intelligence. When discussing intelligence, it is therefore necessary to specify the type of intelligence, thereby implicitly defining the background assumptions and appropriately setting our expectations.

The majority of the formal discipline of Artificial Intelligence has, for the past thirty years, concentrated on the *deliberative reasoning* of human intelligence. It is believed that the same process by which we deliberately reason through, is hypothesized to underlie human intelligent behaviour (Beer, 1990). Traditional Artificial Intelligence has approached the problem of building artificially intelligent systems using the representation and reasoning hypothesis. In this approach, a suitable symbolic representation is used to capture in a given domain knowledge (Goebel, 1988). The given task to be performed must then be expressible as the manipulation of this symbolic representation. A computer program then carries out the required manipulations in such a way as to perform the desired task (Beer, 1990).

Beer (1990) summarizes the traditional AI methodology as follows:

> (1) most intelligent behavior can be modeled on the exemplar of conscious deliberation; (2) deliberative human reasoning is essentially a species of computation over symbolic representations of the world; (3) insights gained from modeling the performance of particular aspects of intelligence in restricted domains will eventually be synthesized into an understanding of generally intelligent behavior in unconstrained interaction with the real world.

Traditional AI has constructed many successful systems which function well in restricted domains of discourse. Typically these systems function exclusively on their own internal representations and for the most part are not temporally constrained when searching their solution spaces. In other words, these systems are not typically real-time systems with time dependant constraints. Search and knowledge are the predominant tools used to solve a given problem with the solutions symbolically represented in a given solution space. It is these traditional AI systems that have, what we will refer to as, *Machine Intelligence*.

1.3 Robotic Intelligence

In an effort to bring these machines into contact with the real world, researchers connected sensors and actuators that would enable their machines to obtain their own input data and to affect the world through their actuators. Brooks (1991) calls this framework, the *sense-model-plan-act* framework, or SMPA for short. Several examples of these mobile robots exist, dating from the late sixties: SHAKEY(Nilsson, 1984) at SRI , the CART (Moravec, 1981) at Stanford and Hilare (Giralt, Chatila and Vaisset, 1984) in Toulouse.

About 1984, a number of people took a different approach to organizing intelligence. Instead of simplifying the world in which their robots lived, as was the case with the SMPA machines, robots were allowed to interact with a highly complex and dynamic environment. This required the robots to be reactive to their environment and operate on time scales similar to those of animals and humans (Brooks, 1991). Agre and Chapman (1987, 1990) claimed that internal symbolic representations were not necessary and could be replaced with simple rules of interaction between an agent and its world. Brooks pioneered much of this work, now termed *behaviour-based* robotics, at MIT by constructing complete robots which operated in dynamic environments using real sensors. Based on this work, it seems possible for a coherent intelligence to emerge from subcomponents interacting in the world without the benefit of explicit internal world models (Brooks, 1986, 1990).

The physical grounding of these systems in the real world and the inherent temporal constraints placed on the robots by the dynamic and unpredictable environment, coupled with physical sensors to provide a representation of the world classify these machines with, what we will refer to as, *Robotic Intelligence.*

Brooks (1991) characterizes a number of key aspects of this style of Robotic Intelligence:

> *Situatedness.* The robots are situated in the world—they do not deal with abstract descriptions, but with the here and now of the world directly influencing the behavior of the system.
>
> *Embodiment.* The robots have bodies and experience the world directly—their actions are part of a dynamic with the world and have immediate feedback on their own sensations.
>
> *Intelligence.* They are observed to be intelligent—but the source of intelligence is not limited to just the computational engine. It also comes from the situation in the world, the signal transformations within the sensors, and the physical coupling of the robot with the world.
>
> *Emergence.* The intelligence of the system emerges from the system's interactions between its components—it is sometimes hard to point to one event or place within the system and say that is why some external action was manifested.

Criticism by its very nature is a destructive act. There is often a cloud of controversy that surrounds behavior-based robotics. The notion of what is important in intelligent systems seems to get lost in the arguments over the need for representation. "The theme common to all this work is that the appropriate patterns of behaviour emerge from the dynamic interaction between an intelligent agent and its environment.

The ability of its internal control mechanisms to somehow mirror the structure of its external environment is irrelevant " (Beer, 1990, p. 14).

Traditional Artificial Intelligence has tried to tackle the problem of building artificially intelligent systems from the top down. Intelligence has been approached through the notions of *thought* and *reason*. Behaviour-based robotics has taken a bottom up approach building systems (e.g., mobile robots), situated in the world, carrying out autonomous tasks (Brooks, 1991). If the latter approach should prove successful then we may expect systems with insect level intelligence to appear first on the path towards human level intelligence. Current systems using this approach show a close similarity to achieving an insect level intelligence (Beer, 1990; Brooks, 1990; Anderson and Donath, 1990). The practical question then becomes: What can we do with them today?

Given that man is not about to create a highly intelligent autonomous robot tomorrow can we make use of these simple intelligent robots we have today? By organizing collections of simple behaviour-based autonomous robots into groups, can useful tasks be accomplished by the collective efforts of these simple machines?

1.4 Achieving Tasks Collectively

Life provides us with countless examples of collective task achieving societies. Bees, Ants and Termites all function collectively in groups, efficiently accomplishing tasks with, seemingly simple, insect intelligence. Researchers have begun to speculate on useful tasks being carried out through collaboration rather than individual effort (see Dario et al., 1991; Yuta and Premvuti, 1991). Steels (1990), inspired by the behaviour of ant colonies, has created a simulation of cooperative behaviour among a group of rock collecting robots. By organizing collections of simple autonomous robots into task achieving groups, it may be possible to reach a new level of robotic intelligence, which is greater than the sum of the simple individuals that comprise the group. We term this new level of intelligence: *Collective Robotic Intelligence* .

In order to build such Collective Robotic systems we will need to develop a control model suitable for controlling populations of task-achieving, behaviour-based, mobile robots. How then should we go about designing the required control systems?

Proposing a Theory. It would seem natural to study one of the examples of existing collective behaviour, namely the social insects. Social insects are a group of arthropods (crusty bugs) studied by entomologists (bug guys). Social insects live in societies and exhibit collective behaviours in maintaining their societies (Wilson, 1971). By studying the proposed theories that attempt to explain the collective behaviour of social insects, we may hope to develop our own control theories and mechanisms that guide us in achieving a collective robotic intelligence.

Our approach to controlling groups of multiple robots is to invoke a common group behaviour. We propose five methods by which this group behaviour may be invoked. The first makes use of a common task and a simple cooperation strategy of non-interference. The second method uses a *follow* behaviour that keeps the group together. The third method involves environmental cues to invoke the same behaviour in each robot. The fourth method allows the robot to invoke its group behaviour once it senses it is within a group. The fifth method invokes group behaviour through autostimulation, a type of broadcast communication.

Testing the Theory. The next step in developing collective robotic intelligence is to test our proposed control theories for collective behaviour. This can be accomplished by simulating the proposed system on a computer. Simulation allows one to test the feasibility of a given control mechanism. We have tested the first two control methods in simulation. However, given the importance of *situatedness* in this style of work and that we ultimately will build these robots, it is very important to simulate only that which we can build. To do otherwise would leave us open to the simplifications so often criticized of simulation work.

Collective behaviour implies a certain cooperative effort is involved in achieving the given task. Therefore, tasks selected for simulation are such that they can not be accomplished by an individual robot, but rather require the collective effort of many individuals. For example, a box cannot be pushed by a single robot either because the box is too heavy or it revolves around its axis. Therefore, in order to push the box some n number of robots will be required where $n > 1$. Simulation allows us to test the control theories in principle, however, the real test takes place in the physical world.

Implementing the Theory. In order to really test ideas of collective robotic intelligence it is important to build complete agents which operate in dynamic environments using real sensors (Brooks, 1990). There has been very little published in the area of cooperation among multiple robots; all of the work to date has been simulation only (see Yuta and Premvuti, 1991 for an example). Working with situated and embodied robots will allow us to observe the dynamic interactions with the environment and other robots. The uncertainties of sensor data and the accumulated errors of such things as wheel slippage can only be ascertained by direct interaction with the real world.

In order to test our first method of control we have designed and built five physical robots. The robots are each equipped with two photovoltaic goal sensors, two infrared robot-avoidance sensors and one stagnation sensor used to provide positive achievement feedback. The robots form a homogeneous group of task achieving autonomous agents capable of collectively moving a box otherwise unmovable by a single robot. They accomplish this task without any explicit communication between the robots. All communication is implicit and in the form of the passive sensing and avoidance of other robots. The experiment demonstrates the feasibility of constructing a simple homogeneous group of reflexive robots to achieve a task collectively.

1.5 Research Goals and Objectives

The methodology outlined above immediately raises a number of questions. Of all the social insects, which should be studied? Is enough known about the collective behaviours of social insects to proceed? If so, what are the predominant theories explaining collective behaviour? In choosing a control architecture for the individual robots, which of the many current approaches should be taken? Is one control architecture more suited for collective behaviour? Or is it necessary to design our own? What sort of simulation tools are required to test and debug these control architectures? In simulation to what level of detail should the robot's sensors and actuators be modeled? How many degrees of freedom of mobility should the robots possess? What types of collective tasks should be simulated? What are the appropriate benchmarks for assessing success or failure? What type of physical sensors will we have to equip our robots with? Is communication among the robots necessary in order to accomplish their task? If so, is that communication explicit or implicit? Most of these questions can only be answered by empirical investigation, by actually building the system and seeing what happens. Like the classical methodology, collective robotic intelligence must ultimately be judged by whether or not it produces models which are successful in illuminating the mechanisms of intelligent behaviour.

The remainder of this thesis describes an experiment in collective robotic intelligence. A robot population simulator called SimbotCity is developed in which any number of robots can be created. The robots are capable of demonstrating several behaviours including wandering, following other robots, herding and goal seeking. Collectively the robots are capable of performing a simple cooperative task designed to demonstrate the feasibility of the approach. The simulated task is then taken into the real world, in which five physical robots have been built. The robots are capable of achieving the same cooperative task as shown in the simulation, namely locating and collectively pushing a large box in their environment. In addition, the robots are capable of switching appropriately between its various behaviours as externally sensed conditions change. The robots are also capable of modifying their course of action if progress is not being made towards the collective task at hand. The methods of achieving collective behaviour have been based in part upon the behaviour exhibited by the social insects.

This work is intended to be an initial exploration into the uncharted world of collective intelligence. As such, its primary goal is to examine the feasibility of the approach outlined above, and to explore some of the initial models for collective behaviour. The work has proven to be formidable, but the experiences of building robots which operate in the real world has been a fruitful endeavor. The robots have shown how a goal oriented behaviour can be coupled with reflexive behaviours to achieve an overall collective task with relatively simple combinational circuits. The next section discusses the outline of the remainder of the thesis.

1.6 Thesis Outline

This thesis represents one of the first known efforts to explore the collective intelligence of a group of situated and embodied mobile robots whose goal is to cooperatively achieve a given task. The remainder of this section provides a brief description of the contents of each chapter.

In this chapter, the foundations of the approach advocated in this thesis are described. We define, what we refer to as, *Robotic Intelligence,* as opposed to the *Machine Intelligence* of the more traditional Artificial Intelligence research effort. We make this distinction to distinguish between Situated and Non-Situated Intelligence. The key characteristics of Robotic Intelligence are outlined with emphasis on the physical grounding of these systems in the real world. The notion of *Collective Robotic Intelligence* is defined and we argue that a control mechanism necessary to achieve collective behaviour may be obtained through the careful study of the social insects. Finally, we provide an overview of the five physical robots we have constructed and the practical knowledge gained from observing them achieve a simple collective task.

Chapter 2 reviews the *behaviour-based* approach to constructing mobile robots. We examine the elements of different control architectures as viewed from their development approach; behavioural control methodologies; behaviour arbitration mechanisms; and communication strategies among individual behaviours. This background is important in order to understand the individuals that comprise the superorganism exhibiting the collective intelligence.

Chapter 3 examines several examples of collective behaviour based on the study of a group of anthropods known as social insects. We primarily concentrate on ants and bees as they represent the most widely researched and understood groups. Social insects have a rich array of sensing capabilities used to invoke behaviour and we examine the many mechanisms involved in producing collective behaviour.

Chapter 4 presents a control theory model for robot populations based on the observations made in the previous chapter. We outline a number of specific observations made from the examples of collective behaviour and speculate how this knowledge may be used in designing our own robot populations. We present our model for group control and outline the steps in designing robots to be used in collective tasks.

In Chapter 5, we present our robot population simulator, *SimbotCity*, and describe its architecture and facilities for testing multiple robots. The physical models used for an individual robot are described as well as the behaviours used to implement collective behaviour. The chapter demonstrates that achieving a collective task is possible by a group of homogeneous behaviour-based autonomous robots without explicit communication among the robots. Finally, a brief overview of related simulation work is discussed

Chapter 6 discusses the implementation of the five physical robots we have constructed. The objectives in building robots is presented along with the collective task we have chosen to demonstrate. The architecture for an individual robot is presented along with a description of the robot's available sensors and actuators. The collective task is presented and a discussion of the resulting demonstrations follows. Finally, the results are compared with the stated objectives and suggestions for other collective tasks are made.

Chapter 7 summarizes the research presented and discusses future research along with proposed methods of implementing group behaviour.

Chapter 2

Behaviour-Based Robotics

2.1 Introduction

Until recently, the field of Artificial Intelligence (AI) has been guided by one unifying idea: the essence of intelligence relied on the symbolic representation of knowledge and reasoning involved the manipulation of this symbolic representation (Chandrasekaran, 1990). This chapter explores another idea called the behaviour-based approach in which there is no, or only partial, representations of the world.

We begin by contrasting the functional approach of traditional AI with that of the behaviour-based approach. Next, we review the behaviour-based approach by dividing the area into three distinct architectures, Non-Representational, Reactive and Distributed Behavioural, in sections 2.3, 2.4 and 2.5, respectively. Finally, we provide a summary of the behaviour-based approach.

2.2 Functional Versus Behavioural Architectures

The majority of the work in AI has followed the Turing dream of capturing intelligence as a disembodied representational system, which we refer to as machine intelligence. Researchers interested in creating a robot that interacts with the real world have made the assumption that such a robot would also possess machine intelligence. These robots would contain sensors at one end and actuators at the other end. Their controllers were referred to as functional based controllers and were based on machine intelligence (Chandrasekaran, 1990).

A functional based controller is a functional decomposition resulting in modules each of which performs one stage in a processing pipeline (Figure 1.). Sensor data is fed into the left end of the pipeline with the first module designed to do sensor processing. This data is then used as input to a module designed to produce a central world model. A planner module uses this world model to produce a plan which is passed on to an execution module which ultimately controls the actuators or effectors. This architecture is illustrated in the following diagram.

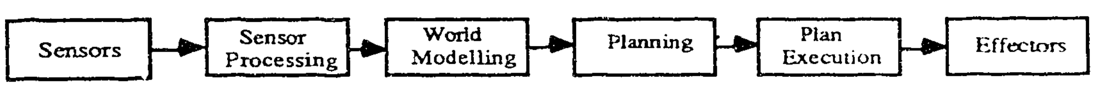

Figure 1: Functional based controller with modules wired in series.

This approach is very sensitive to failures in any one of the modules; like the old serially wired Christmas lights, malfunction of any one bulb causes the whole string to go out.

A behaviour-based controller is organized into task achieving behaviours. In this approach the controller is divided into modules, one for each type of behaviour to be exhibited by the system. Examples of such behaviours are seek-light, avoid-darkness or avoid-obstacles. In this approach sensors are inputs to all modules, or behaviours, and all modules output to the effectors. This architecture is illustrated in the following diagram

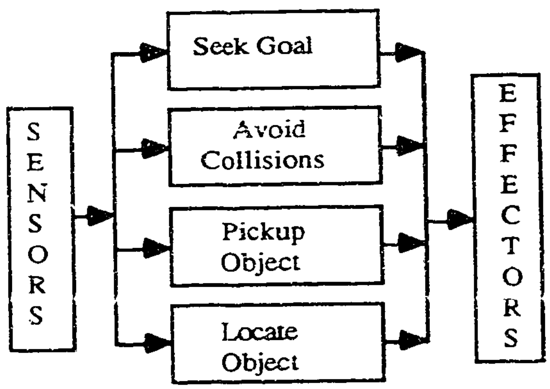

Figure 2: Behaviour based controller with modules wired in parallel.

This control architecture is more robust and less sensitive to failures in any of the modules. This robustness is due to the parallel manner in which the modules are wired. Failure of a behaviour will result in a diminished, but a still functioning, robot. Like today's parallel wired Christmas lights, failure of any bulb still keeps the string lit.

Behaviour-based control can be broadly divided into three approaches: First, there is the non-representational or connectionist approach. Proponents of this approach argue that intelligent action does not require or use explicit representations and their processing, and behaviour can be achieved by reflexive actions. Braitenberg (1984), Beer (1990), Beer et al. (1990a), Travers (1988), Coderre (1988), and Sekiguchi et al.

(1989) are examples of this approach. The second is a reactive approach in which perception-directed *reactive* actions are used as a way of responding to a complex environment without complex planning. Responses are indexed directly over the situation description, rather than resulting from complex problem solving using abstract world models. Sensors constantly monitor the changes in the world and additional reactive steps are taken where appropriate. The work of Agre and Chapman (1987, 1990), Kaelbling (1987), Kaelbling and Rosenschein (1990) and Anderson and Donath (1988a, 1988b, 1990) are examples of this approach. The third approach is a combination of the first two with the addition that action generation may not be performed centrally at all. In this approach reactiveness of responses is combined with distribution of action-generation. Brooks (1986, 1990), Payton (1986, 1990a), Payton et al. (1990) and Arkin (1987, 1990) are examples of this approach. In the next section we examine the non-representational approach.

2.3 Non-Representational Architectures

This section looks at the non-representational or connectionist approach (also referred to as reflexive) to behavioural robot control. This approach produces *reflexive* behaviours. In section 2.3.1 we define reflexive behaviour, and examine several examples of this approach. In section 2.3.2 we consider the work of Valentino Braitenberg and his series of robots known as *vehicles*. In section 2.3.3 we consider Michael Travers' work on *Animal Construction Kits*, an idea inspired by Braitenberg's vehicles. In section 2.3.4 we look at Bill Coderre's *Petworld*, in which the behaviour of "pets" is controlled by hierarchical experts inspired by Minsky's Society of Mind Theory (Minsky, 1985). In section 2.3.5 we consider the work of Randal Beer and his *artificial insect* inspired by a close study of the neural circuit of a simple insect. In section 2.3.6 we examine the connectionist approach of Sekiguchi et al. in which behaviours are generated by a trained neural network. Finally, we conclude with a discussion in section 2.3.7 about the sufficiency of reflexive behaviour alone to define complex task achieving behaviour.

2.3.1 Reflexive Behaviour

Researchers using non-representational approaches argue that much of intelligent action does not require or use explicit representations and their processing. "Connectionism has been embraced warmly by many philosophers on the grounds that it provides such a non-representational account of cognition" (Chandrasekaran, 1990). This approach produces very reflexive behaviour. Anderson defines *reflexive* behaviour as "the response of the robot at time $t+1$ is completely determined by a specific set of stimuli at time t and is independent of other unrelated external/internal events" (Anderson & Donath, 1988a). Reflexive behaviour differs from *reactive* behaviour. Reflexive behaviour has no memory and s a fixed relationship between the stimulus and its response. Reactive behaviour, on the other hand, has memory and as a result produces behaviour in which the stimulus/response relationship is not necessarily fixed but may also depend upon other external/internal environmental factors, unrelated to the stimulus. In the next section we consider the reflexive behaviour of Braitenberg's Vehicles.

2.3.2 Vehicles

Braitenberg proposes a series of 14 robots known as *vehicles* (Braitenberg, 1984). The 14 vehicles represent a series of hypothetical, autonomous robots that exhibit increasingly sophisticated simple behaviour such as seek-light and avoid-obstacles. Each of the robots in the series incorporates the essential features of its predecessors with some incremental improvement in behavioural complexity. Of the 14 vehicles, the first six represent a completely reflexive approach to robot control.

The simplest vehicles are created using a stimulus/response paradigm and four connection types: excitory (+) or inhibitory (-) connections, and crossed or uncrossed connections, where crossed refers to connecting a sensor to the motor on the opposite side. The result is the creation of four vehicles illustrated below.

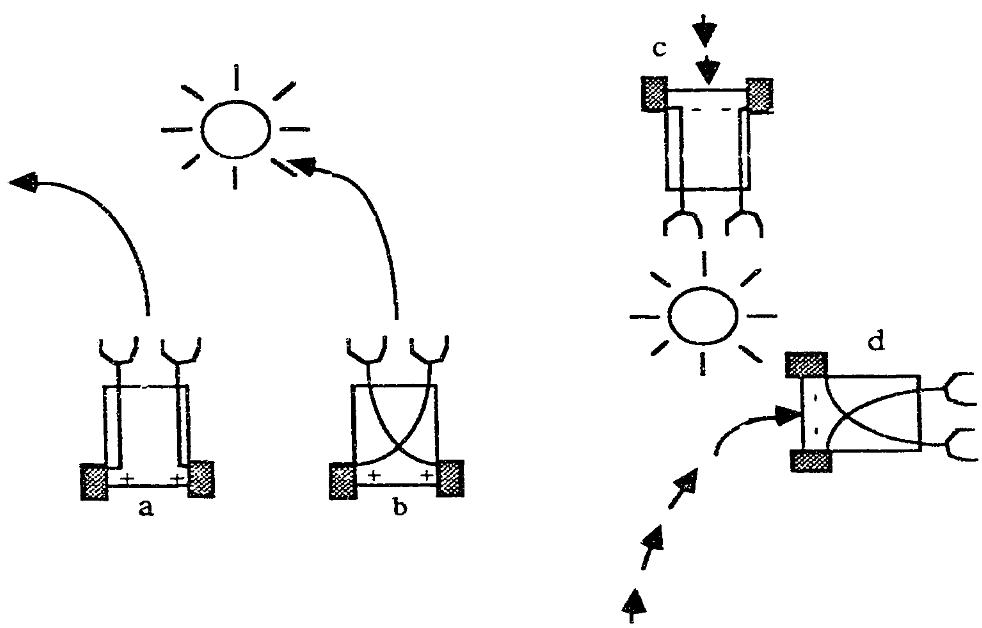

Figure 3: Braitenberg's vehicles. Sensors are connected to motors with either crossed or uncrossed, excitory or inhibitory connections. Vehicle 3b orients toward the source, 3a away from it. Vehicle 3c and 3d with inhibitory influence of the sensors on the motors.

All vehicles in figure 3 have simple behaviours as a result of these connections. In vehicle 3a each sensor is connected uncrossed to a wheel motor causing the vehicle to turn away from a source. For example, the right sensor activates the right wheel motor causing the vehicle to turn towards the left. In vehicle 3b the sensors are cross connected causing the sensor to activate the wheel motor on the opposite side; this causes the vehicle to move towards a source. Vehicles 3a and 3b both DISLIKE sources. Vehicle 3a tends to avoid them by escaping to a place where the source is hardly felt. Braitenberg calls vehicle 3a a COWARD. Vehicle 3b is also excited by the presence of sources, but turns towards them and hits them with high velocity, as if it wanted to destroy them. Vehicle 3b is therefore AGGRESSIVE (Braitenberg, 1984).Vehicles 3c and 3d have a inhibitory connection between the sensors and the motors. This will allow the vehicles to slow down in the presence of a strong stimulus and speed up when the stimulus is weak. Vehicle 3c is said to LOVE the source and will come to rest in the presence of the source. Braitenberg describes vehicle 3d as an EXPLORER since it too likes the source but will come to rest pointing away from it ready to leave as soon as another weak source is detected. Vehicles 7 - 14 acquired a sense of memory, allowing them to behave differently given different environmental considerations, and become increasingly reactive.

2.3.3 Animal Construction Kits

Travers creates an *animal construction kit* which allows the user to assemble active artificial animals from prefabricated components (Travers, 1988). Inspired by Braitenberg's proposal, Travers has constructed a simulation environment which allowed point-like robots to interact with food and obstacles. Travers designed two systems: the first, *Brain Works* uses a neural model of computation. The second system, *Agar*, uses agent-based computation inspired by Minsky's Society of Mind Theory (Minsky, 1985).

Brain Works allows the user to construct a reflexive nervous system for a simple animal. The animal is equipped with several sensors and motors for movement. The sensors include eyes and touch bumpers which respond respectively to food and to obstacles in the animal's world (Travers, 1988). The animal's basic task is to catch food while avoiding being blocked by a wall or obstacle. Travers extended Brain Works by allowing evolved animals. Survival and reproduction of these evolved animals were based on maintaining an energy reserve by finding and eating prey. An energy reserve was depleted by movement and the passage of time, and increased by eating prey. This extension evolved the system from reflexive to reactive behaviour by replacing the neural behaviour model with a matrix that defines a function from the vector of sense inputs to the vector of motor outputs.

2.3.4 Modeling Behaviour in Petworld

Coderre has created a simulation environment called *Petworld* for modelling aspects of animal behaviour. Unlike Brain Works, behaviour in Petworld is modeled as a rigid hierarchy of simple agents that make recommendations to their superiors (Coderre, 1988). These recommendations are in the form of rankings of alternatives. Conflicting behaviours are resolved by agents either choosing, compromising or displacing.

A pet's control system is a hierarchy of modules called *experts*. Each expert makes recommendations by providing an output consisting of a *ranking*—a list of possible actions with numeric weights attached—telling how good those actions are in the opinion of the expert (Coderre, 1988). The pet executes the highest ranking action recommended by the topmost expert. This approach is very similar to the connectionist approach of assigning weights to the inputs of neurons during training. In the pet control architecture a partial order is assigned to each expert's behaviour. For example, finding food and interacting are most important, followed by nest building, exploring and homing into the nest. To resolve conflicting behaviours say between foraging and combat, a compromise is made, dependent on how serious the danger of starvation or attack is (Coderre, 1988). This behaviour could then be classified as reactive by the previous definition since the behaviour exhibited by the pet will depend on both internal (starvation) and external (attack) events.

2.3.5 An Artificial Insect With Adaptive Behaviour

Randal Beer has created an artificial insect capable of demonstrating simple adaptive behaviour (Beer, 1990). This has been achieved through the careful study and simulation of the biological mechanisms underlying the autonomous behaviour of simpler natural animals (Beer et al., 1990a). They call their approach *Computational Neuroethology,* since Ethology is the study of the behaviour of animals in their natural environments (Lorenz, 1981), and Neuroethology is the study of the neural mechanisms underlying this behaviour (Camhi, 1984).

The behaviour of the simulated insect is controlled by an artificial nervous system and is based in part on specific neural circuits in several natural animals. Model neurons are interconnected by weighted synapses through which they can inject current into one another (Beer et al., 1990a). In this respect Beer's model is similar to several neural models that have previously explored in the field of artificial neural networks (for an example see Hopfield, 1984). Behaviour arbitration is handled by an explicit ordering of the behaviours. Beer's artificial insect's behavioural repertoire will increase as the field of Neuroethology makes new discoveries on the mechanisms that control behaviour.

2.3.6 Behavioural Control Using Neural Networks

Sekiguchi et al. (1989) have constructed a physical robot whose behaviour is governed by patterns in sensor data detected by a trained neural network. A total of twelve sensors are used for detecting both internal and external changes in the environment. This data is then fed to a hierarchical network which has been previously trained. Both the input vector from the sensors and output vector to the motors are boolean (0 or 1). Short term memory provides a hysterisis effect on the input sensor data. This allows the robot to continue executing a behaviour pattern once the stimulus is removed. Training is accomplished through back propagation of the error term from the output layer to the input layer.

2.3.7 Discussion

It can be argued that connectionism is as representational as the classical symbol manipulation systems, the main difference coming from the type of representation (Chandrasekaran, 1989). The advantages of the simple reflexive behaviours, exhibited in Braitenberg's vehicles 1 - 6, and Travers unevolved animals, is the simplicity of control architecture used to create the behaviors. However, it can be seen, in Braitenberg's later vehicles, Travers evolved animals, Coderre's Pets and Sekiguchi's robot that more complex behaviours require the introduction of some internal model achieved through the introduction of memory. This is implemented in Braitenberg's vehicle 7 by using special wire called Mnemotrix, and in Travers' animals by replacing neurons with a matrix memory. It is also present in Coderre's pets in the form of experts providing ranked output actions and Sekiguchi's short term memory. In each of these systems we witness an evolution of increasingly complex behaviour that begins as purely reflexive behaviour and evolves toward reactive behaviour. The above examples seem to suggest that reactive behaviour and memory are required for a richer expression of robot behaviour. This is also the question asked by Anderson and Donath (1988a):

> The issues with which we are concerned are fundamental; can autonomy result from simple reflexive stimulus/response forms of behavior in which there exists a rigid relationship between a specific stimulus and exhibited response, or is reactive behavior required?

In a simulation experiment on location directed open space wandering performed by Anderson and Donath (1988a) a combination of four reflexive behaviours with no arbitration mechanism or memory for the behaviours produced a cyclical behaviour due to the lack of internal state within each behaviour.

We have also experienced this cyclical behaviour in experiments with our reflexive robot Herbie. Herbie has three binary light sensors, two forward looking and one vertical looking, and one each left and right wheel motors. Endowed with two simple reflexive behaviours, *seek-light* and *avoid-darkness*, we have observed the robot to enter a cyclic behaviour pattern, of forward - reverse motion, when entering a dark region at slow speeds shown in figure 4.

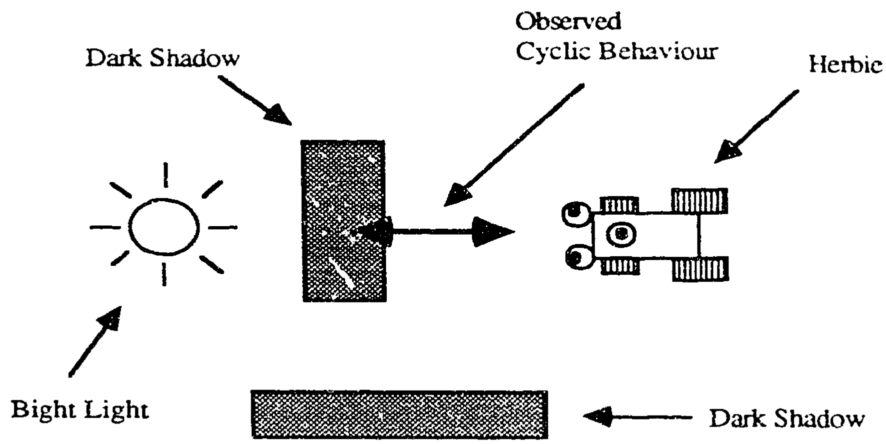

Figure 4: Herbie - the photovore is a reflexive light seeking, dark avoiding robot. Note the cyclic behaviour as the robot enters and then backs out of the darkness. Without memory, this behaviour is repeated continuously.

This experiment verified to us the same conclusion reached in the simulation by Anderson and Donath (1988a), that an autonomous system requires reactivity and memory. This approach to behaviour based control is examined in the next section.

2.4 Reactive Architectures

This section examines the *reactive* approach to behavioural mobile robot control. In section 2.4.1 we will define reactive behaviour and provide an overview of the approach. Next we consider three examples of this method of autonomous mobile robot control. In section 2.4.2 we examine the work of Agre and Chapman (1987, 1990) in their implementation of Pengi, a program which controls the behaviour of a simple penguin embedded within a commercial video arcade game known as Pengo. In section 2.4.3 we consider the work of Kaelbling (1987) and Kaelbling and Rosenschein (1990) and their architecture for reactive systems. In section 2.4.4 we examine the work by Anderson and Donath (1988b, 1990) in constructing reactive behaviours from a collection of reflexive behaviours. Finally, we conclude with a discussion in section 2.4.5 on the merits of reactive behavioural control for an autonomous mobile robot.

2.4.1 Overview of Reactive Behaviour

Reactive behaviour has been previously described as behaviour in which the stimulus/response relationship may depend on the occurrence of specific events. For example, presenting your hungry pet with food may invoke a strong response if some time has passed since its last feeding, however, once satiated the same food stimulus is unlikely to invoke the same response because the pet's internal state has changed. The primary difference between reflexive and reactive behaviour is memory; reactive behaviour has a temporal ordering which affects the stimulus/response relationship whereas reflexive behaviour is without memory and therefore, has a fixed stimulus/response relationship. In the next section we consider just such a reactive approach.

2.4.2 Pengi: An Implementation of a Theory of Activity

Agre and Chapman began to question the supposition that action derives from the execution of plans within the framework of problem solving and reasoning with representations. They believed that activity could be derived from simple machinery interacting with the immediate situation (Agre & Chapman, 1987).

To test their theory, they built a program called Pengi which controlled the behaviour of a simple penguin embedded within a commercial video arcade game known as Pengo. In Pengo bees chase penguin's and if caught the penguin dies.

Penguins and bees live in a two dimensional maze consisting of walls made of slideable ice blocks. Penguins can kill bees, and likewise bees kill penguins, by kicking an ice block into a bee or penguin.

As an example of how activity results without planning, but instead relies on constant evaluation of the immediate situation, consider the implementation of their situation-action like reactive rules. Imagine the situation of a penguin being chased by a bee along a corridor. The penguins behaviour is the result of two simple rules, or reflexes: (R1) when you are being chased, run away; (R2) if you run into a wall, kick through it. They argue that the penguin's behaviour is not governed by any preconceived idea of what will happen, but rather as a result of the current situation (i.e., bees chasing penguin). This allows the behaviour to be opportunistic, and nerefore robust under uncertainty. The simple nature of the reflexive behaviour of R1 and R2 allows for real-time reactive activity. Behavioural arbitration is handled by a hierarchical behaviour selection mechanism. In the next section we consider an alternative reactive architecture.

2.4.3 An Architecture For Intelligent Reactive Systems

Kaelbling takes a top down approach to behaviour construction. The desired behaviour of the robot is specified by constructing a program in terms of a top level goal and a set of goal reduction rules which leads to the goal. The proposed architecture is divided into two components: perception and action. The perception component is horizontally subdivided into several layers of abstraction with uninterpreted sensor readings available at the lowest level and world models available at the highest level. The action component is also horizontally subdivided into a set of behaviours. The action component receives input from the perception component and outputs commands to both the sensors and effectors as illustrated in figure 5 from Kaelbling (1987):

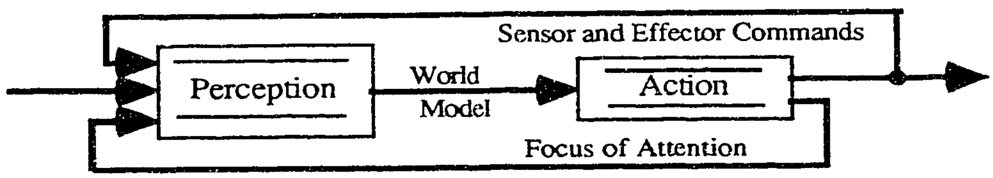

Figure 5: Kaelbling's Top Level Decomposition.

This allows for an action directed perception and is unique among many other systems. The system is reactive through the use of an incremental planner that immediately notices changes in the environment before the completion of the plan and adjusts by starting a new plan with the given situation in mind.

The robot's control architecture is a hierarchical decomposition of behaviours. Arbitration amongst conflicting behaviours is handled by procedures called *mediators*. In the next section we examine a method of assembling simple reflexive behaviours into a system that provides reactive control.

2.4.4 Organizing Reflexive Behaviours for Reactive Control

Anderson and Donath were interested in investigating the requirements for autonomy. They began by looking at behavioural patterns that were purely reflexive (Anderson & Donath, 1988a). They came to the conclusion that while it was possible to construct autonomous behaviour in terms of a set of fixed primitive reflexive behaviours, the degree of autonomy achievable may be limited (Anderson & Donath, 1988b). They also concluded the limitations of reflexive behaviour were due to the lack of memory of previous events.

Their approach to creating reactive behaviours was bottom up. Lower level behaviours were constructed from individual reflex behaviours. High level behaviours were constructed using the previously defined lower level behaviours as components and devices called a goal detector. These goal detectors acted as boolean event detectors with inputs from sensors. Thus these higher level behaviours act as arbitrators of the component behaviours and events in the environment. The net resulting behaviour of the mobile robot is reactive; the set of behaviours active at any point in time is a function of its location in the environment (Anderson & Donath, 1988b). In the next section we discuss the advantages and disadvantages of the reactive approach

2.4.5 Discussion

Situatedness (reactive behaviour that results from an agent responding to a given situation) behaviour, of the type exhibited in Pengi, provides a fast response demanded by real-time environments but lacks a goal-directed component thought to be a requirement for intelligent autonomous systems. Like Simon's Ant (Simon, 1970), seemingly intelligent behaviour is more a result of the complexity of the environment than the complexity of the intelligent agent. That is not to say that this approach does not merit further study. On the contrary, we believe that the study of intelligent robotic behaviour should begin bottom up with an understanding of the simpler forms of intelligence. Goal directedness belongs to the higher levels of intelligent behaviour that result from willful motivation and will undoubtedly be required in an intelligent autonomous mobile robot.

A common question found in all the behaviour based approaches is how to provide arbitration amongst the competing behaviours. The weakness in Agre and Chapman's approach revolves around their behaviour arbitration mechanism. The designer must understand the dynamics of the penguin's environment and hardcode the arbitration mechanism into the rules of action. This does not provide a general method for behaviour arbitration.

In Kaelbling's approach behaviour arbitration is handled by *mediators* who receive input from behaviours and sensor data and decide which behavioral pattern to follow. This is very similar to Anderson and Donath's approach with higher level behaviours accepting inputs from lower level behaviours and *event detectors* which monitor the environment. The difference being Kaelbling takes a top down approach by specifying a desired behaviour for the robot and using goal reduction rules which lead to the goal. This allows their action agent to model any type of conditional or heuristic strategy in which a fixed relationship between the agents inputs and outputs. Anderson and Donath, on the other hand, use a bottom up approach by defining new behaviours in terms of previously defined behaviours. The use of abstraction in this approach is more powerful and keeps the details of a particular behaviour hidden. This allows for a complete decoupling of the behaviours and separates the mechanism for controlling the behaviours from the behavior's implementation details. This approach allows for an explicit representation of complex forms of behaviour through the use of abstraction. The problems with Anderson and Donath's approach lie in the reflexive behaviour components method of response. Each primitive reflexive behaviour transforms a set of stimuli into a response. The response of each behaviour is expressed in the form of a potential field (Anderson and Donath, 1988b). The potential field method suffers from local minima in the net potential field which results from combining the potential fields of two conflicting behaviours. The robot will then appear "stuck"; a problem referred to as "stagnation". This could be resolved through the use of a behaviour which monitors the robot's progress and envokes an alternative strategy when the currently active behaviour is found to have stagnated.

Any intelligent autonomous mobile robot operating in a changing dynamic environment will have to be reactive in order to operate in real-time. The open question for such a system is how to combine and provide a method with which to arbitrate the behaviours. In the next section we examine the third approach to behaviour-based control of a mobile robot in which action generation may not be performed centrally at all.

2.5 Distributed Action Generation

This section examines the third approach to behaviour-based control for an autonomous mobile robot. This method combines some aspects of the first two methods with the distribution of action generation. The central theme for this section will consider the distribution of action generation. Some approaches do not make use of internal representations (Brooks, 1986) and some do (Arkin, 1987; Payton, 1986). We begin in section 2.5.1 with an overview of the approach. In section 2.5.2 we consider the work of Brooks and his method without representation. In section 2.5.3 we examine Arkin's approach to combining reactive response with the more classical AI planning systems. In section 2.5.4 we look at the work of Payton, both his earlier work with reactive systems and his newer approach which incorporates a connectionist model for behaviour selection (Payton, 1990). Finally, we conclude with a discussion in section 2.5.5 on the merits of the above three methods.

2.5.1 Overview Distributed Action Generation

Methods of distributed action generation usually feature concurrent asynchronous decision making processes or behaviours. Centralized systems responsible for world models usually must engage in computationally expensive sensor fusion. The distributed approach allows for real-time reaction necessary in constantly changing environments. In all approaches some mechanism must be provided to combine the outputs of multiple behaviours. We consider the first of these approaches in the next section.

2.5.2 The Subsumption Architecture

Brooks proposed a new architecture for controlling a mobile robot called the subsumption architecture (Brooks, 1986). The controller is built as a series of layers each designed to exhibit increasing levels of competence. Each layer is composed of a task achieving behaviour. The controller is built incrementally with new layers being added on top of previous layers. The new layer leaves most of the previous layer's behaviour intact but can take control or "subsume" some aspect of the previous behaviour. In this fashion the robot begins with a working controller. As new layers are added, on existing debugged behaviours, the robot improves in competence. The control of the robot is accomplished without any internal representation of the world.

Each layer of task achieving behaviour is composed of several modules which are finite state machines with a set of inputs and outputs. Modules can communicate by sending messages along their output line. Since there is no handshaking, message delivery is not guaranteed. All modules can read any of the robot's sensors and can theoretically manipulate the robot's effectors; although, generally only the lowest layer behaviour controls the effectors directly

Higher level behaviours control lower level behaviours through *suppress* and *inhibit* links. Messages sent on an inhibit link stop all messages on the target link for a fixed period of time. Messages sent on a suppress link replace the messages on the target link.

Behaviour arbitration is handled by a fixed priority scheme just as in Kaelbling's arbitrator mechanism. Later work by Cudhea and Brooks (1986) extended this architecture to provide a different mechanism for resolving conflict between behaviours. This was implemented as *difference engines* designed to reduce the difference between the actual world state and the desired state. To combine different behaviours a simple timeout mechanism is used to control the transition between behaviours. Jon Connell provides several enhancements to Brooks' approach (Connell, 1990). Based on the subsumption architecture Connell's approach differs from that of Brooks in the manner in which the control system is layered. In Brooks' approach layers define a total order on the behaviours of the robot, while Connell defines a tree-like partial order. Connell also allows the priority to vary among the layers. Other differences include less state and more modularity. In the next section we consider an approach of combining reactive distributed control with the more classical AI planning approach.

2.5.3 Motor Schema Based Control

Purely reactive systems are incapable of formulating and following longer-term goals because they are always immediately reacting to the world. This has led a number of researchers to propose various schemes that integrate reactive responses with more classical AI planning techniques which reason about explicit internal models of the domain (Beer, 1990).

Arkin has proposed one such method called *schemas*. A schema can be viewed as a generic specification of a computing agent. Motor-schemas correspond to a primitive behaviour that can be combined with other motor schemas to yield a more complex behaviour (Arkin, 1987). For each motor schema activated, perceptual schemas are created to detect events. If such an event occurs, a new motor schema is instantiated as a concurrent process. Schemas communicate through a blackboard ensuring a high degree of concurrency. Arkin's motivation for the use of schemas comes from neuroscientific, psychological, and robotic sources (Arkin, 1990).

Instantiated motor schemas drive the robot to interact with its environment. At the highest level they are used to satisfy a goal created by the planning system. At the lowest level they provide primitives to the robot. The motor schema based control system is discussed within Arkin's Autonomous Robot Architecture (AuRA). AuRA has a hierarchical planner consisting of a Mission Planner, a Navigator, a Pilot and Motor Schema Manager. The Pilot is responsible for implementing a path developed by the Navigator. To do so, the pilot chooses from the available sensing mechanisms and motor schemas. These are passed on to the Schema Manager for instantiation. Distributed cor[illegible] occurs within the Manager. Schemas are instantiated with parameters, so two generic schemas can have two different instantiations. Arbitration amongst behaviours is handled through the use of potential fields. Thus, motor schemas produce a field (vector) which is summed along with other motor schemas vector outputs. As previously mentioned the potential field method has problems with local minima. To over come this Arkin introduces a background noise schema which produces a low-magnitude random direction velocity vector sufficient to change the robot's position if it becomes stuck at one of these local minima. In the next section we examine another multi-behaviour approach which later introduces a novel link to the first connectionist approach previously discussed.

2.5.4 Plan Guided Reaction

Payton (1986) describes a multi-behaviour approach for reflexive control of an autonomous robot. Virtual sensors (similar to event detectors) were used to recognize features in the environment and send this data to reflexive behaviours which in turn sent motor commands to a blackboard. A control module, using priority arbitration, selects one message and sends it to the motor actuators. This is similar to Arkin's schema based approach. But no mechanism existed to combine concurrent behaviours. This system led to very tight coupling between sensing and action. The overall system was hierarchical instead of layered but the lowest level of control was very similar to Brooks' subsumption architecture (Payton et al., 1990). At the lowest level an action was determined by output of multiple concurrent, independent behaviours. This distributed approach allowed them to avoid the computationally expensive task of generating a centralized world model. This allowed for a quick reactive response instead of having to deal with the delays imposed by sensor fusion. They termed this command fusion.

Although the above approach led to many successes they gained most from a close examination of how it failed. The main result was their discovery of the problems abstraction caused by limiting access to critical information. This discovery led to the proposal of a new architecture designed to overcome this limitation (Payton, 1990).

The new architecture eliminated abstraction wherever possible. In order to eliminate the communication problem between behaviours they made behaviours as fine grain as possible by eliminating internal states and instance variables. These simple decision-making units and their interconnection collectively define a behaviour. This allowed them to move away from a priority-based arbitration mechanism to a distributed one in which arbitration is carried out by the fine-grained connectionist architecture. Behaviours can then explicitly indicate their bias in regards to the available choices. The weighted output from a behaviour is used as one input to a multi-input command unit. The command unit then selects the behaviour with the greatest weight. Additional behaviours could be added and would influence the decision-making by inputing their preferences to the command units (Payton , 1990).

2.5.5 Discussion

The Brooks' approach combines the non-representational approach with the reactive approach and provides a non centralized action generation. However, it lacks modularity as one behaviour is not protected from the details of others. Connell proposes a solution by keeping modules completely independent (Connell, 1990). Hartley proposes a solution by introducing a new abstraction called a control connection (Hartley, 1991). Each control is associated with an *arbitrator* whose job is to select behaviours based on a priority list.

Brooks method also forces a fixed priority on the behaviours thus lower level behaviours can not control (subsume) higher level behaviours. Sometimes a low level should be able to override a higher level, consider what happens when you touch something hot. The spinal cord takes over eliminating the danger while the higher level inputs are ignored (Hartley, 1991). However, this response could be implemented as an emergency behaviour which takes control of the low level in these situations.

Lack of modularity also causes a problem if lower levels are changed. Since higher levels are dependent on the inner workings of lower levels, changes cause large redesigns of the controller.

Arkin's method differs from that of Brooks in that there is no layering in schemas, instead it is a collection of networked autonomous agents changing as the perception of the robot changes. The method uses potential fields and can suffer from local minima.

Payton's previous approach was non-connectionist and they attribute the success of their current connectionist fine-grained approach to the elimination of abstraction. We don't feel this is quite correct in the sense that it is not the elimination of abstraction that is important but rather the elimination of internal state within the individual behaviours. Abstraction is a powerful tool in constructing behaviours. Payton's *command units* are equivalent to Kaelbling's *arbitrators* or; Cudhea and Brooks' *difference-engines* ; or Anderson and Donath's higher level behaviours and *event-detectors*.

2.6 Summary

In this chapter we have surveyed a cross section of the behaviour-based mobile robot control systems. We have presented the work in the three broad categories of Non-Representational or Reflexive Architectures, Reactive Architectures and a combination of the two in the Distributed Behavioural Architectures (for a good collection of papers see Maes, 1990). The approaches may be summarized in the following table:

Control Method	Arbitration	Communication	Representation
REFLEXIVE			
Braitenberg (1984)	Weighted Inputs	Blackboard	Raw Sensor Data
Beer (1990)	Weighted Inputs & Fixed Priority	Blackboard	Raw Sensor Data
Sekiguchi (1989)	Hierarchical Net.	Mem/Blackboard	Sensor Data/Mem
Travers (1988)	Weighted Input	Blackboard	Raw Sensor Data
Coderre (1988)	Hierarchy Agents	Message Passing	Sensor Data/Mem
REACTIVE			
Agre & Chapman (1990)	Hierarchy/Fixed	Message Passing	Raw Sensor Data
Kaelbling (1987)	Mediators	Message Passing	Sensor Data/Maps
Anderson & Donath (1990)	Hierarchy/Potential Field	Message Passing	Raw Sensor Data
DISTRIBUTED			
Brooks (1990)	Fixed Priority	Message Passing	Sensor Data/FSMs
Arkin (1990)	Potential Field	Blackboard	Sensor Data/Maps
Payton (1990)	Fixed/Weighted	Blackboard	Sensor Data/Maps

Table 1: A summary of the behaviour-based approaches.

Behaviour-based control systems are a relatively new and promising way in which to design autonomous intelligent robots which are situated in the real world. All the systems presented are *single* task achieving autonomous robots with Robotic Intelligence that differs significantly from the Machine Intelligence found in today's classical Artificial Intelligence systems. The difference lies in the representational requirements of a situated and embodied intelligent agent. Those requirements call for the representation to be expressed as immediate sensor data indexed directly off the environment.

Is it possible to organize such machines into groups of task achieving autonomous robots? And how would we go about controlling them? In the next chapter, we try and answer these questions by examining one of the natural occuring examples of collective behaviour: The social insects.

Chapter 3

Collective Insect Intelligence:
An Example of Collective Behaviour

3.1 Introduction

Can Nature provide us with hints in designing control theories for collective behaviour? One of the most challenging questions in science is how the behaviour of large systems is generated from their individual components. By studying Nature's examples, valuable lessons in population dynamics and its control may help us to create our robotic society.

In order to build such a *population* of cooperating robots we will need to develop a control model suitable for controlling groups of task-achieving, behaviour-based, mobile robots. How then should we go about designing the required control system? One possible approach would be to study an existing example of collective task achieving behaviour found in the social insects. Social insects live in societies and exhibit collective behaviours in maintaining their societies (Wilson, 1971). Theraulaz et al. (1991) and Deneubourg et al. (1991) have both proposed models with which to control groups of interacting robots, based on studies of social insects. Parallels between artificial self-organizing systems and insect societies are mentioned in Deneubourg and Gross (1989), Deneubourg et al. (1987), Moyson and Manderick (1988) and Steels (1987, 1989).

In this chapter we examine naturally occuring examples of a control theory for insect populations. We begin by examining several examples of collective behaviour found in the social insects.

3.2 Insect Societies: An Example of Collective Behaviour

Social insects are a group of arthropods (crusty bugs) studied by entomologists (bug guys). Social insects live in societies and exhibit collective behaviours in maintaining their societies. A society is "a group of individuals that belong to the same species and are organized in a cooperative manner" (Wilson, 1971, p. 5). The insect colony is often referred to as a superorganism because many of the social phenomena it displays are similar to the physiological properties of organs and tissues. However, the holistic attributes of the superorganism occur in a behavioural way as a result of the simple repertories of individual colony members, and are more easily understood than the molecular basis of physiology (Wilson, 1971). Information on social insects concentrates on four main species: ants, termites, and the more highly organized bees and wasps. Wilson distinguishes these insects as a group by their common possession of three traits: "individuals of the same species cooperate in caring for the young; there is a reproductive division of labour, with more or less sterile individuals working on behalf of fecund individuals; and there is an overlap of at least two generations in life stages capable of contributing to colony labour, so that offspring assist parents during some period of their life" (Wilson, 1971, p.4). The most studied and understood groups are ants and bees. In this section we will concentrate on these two groups by examining the elements of collective behaviour: sensor capabilities, mental capabilities, examples of collective behaviour, and the role communication plays in collective behaviour. We begin with an overview of ants and bees.

3.2.1 Ants and Bees: An Overview

Ants are considered the premier social insects. They are also the most abundant of the social insects. At any given moment there are 10^{15} ants living on the earth (Wilson, 1971). Ants are organized through a caste system. There are three castes among the female: worker, soldier, and queen. The males form a caste by themselves but only in the loosest definition of the term. Ants have specialized workers designed to perform different tasks. For example, larger workers do most of the foraging while smaller workers are responsible for brood care.

Odour trails are the principal means of communication among ants. Single scout workers are able to recruit a large number of their nest mates to food finds in a matter of minutes by means of odour trails (Peacock, Waterhouse and Baxter, 1955).

Behaviours seem to be triggered by externally sensed events. For example, the swarming behaviour can be triggered by the overcrowding of the workers in the nest chambers (Peacock et al., 1950). Swarming behaviour also begins upon the detection of the light of dawn (Schneirla, 1956). The same behaviour will begin at a later time on overcast days. Ants usually return to their nest at dusk. Behaviour is also triggered by movement and scent of their prey (Schneirla, 1956). The array of different sensing capabilities enables ants to display a large collection of different behaviours.

The honeybee (*Apis mellifera*) has been the most intensively studied of all the social insects (Wilson, 1971). They have a simple caste structure consisting of worker and queen. Bees have been found to exhibit a temporal division of labour (i.e. as bees age their tasks change) which is thought to depend on a rigid time schedule of glandular development (King, 1933; Lindauer, 1952).

Bees have an advanced visual and auditory system of communication. The waggle dance "is a ritualized reenactment of the outward flight to food or new nest sites; it is performed within the nest and somehow understood by the other workers in the colony" (Wilson, 1971, p. 94). The workers are then able to translate this dance into a unrehearsed flight of their own. The meliponine bees are also able to interpret modulated symbols. They transmit sound signals which correlate in duration and frequency with the distance of food finds (Wilson, 1971).

Like the ants, behaviour can also be triggered by externally sensed events. For example experiments by Schricker (1965) determined that honeybees use the three ocelli, which are simple eyes, to monitor light intensity. The intensity of the light is used by worker bees to govern the amount of time used for food collecting behaviours. Bees with their ocelli blinded began collecting food much later in the morning and ceased flying earlier in the evening than unoperated workers.

Next we examine some of the elements involved in producing behaviour in both ants and bees.

3.2.2 The Elements of Collective Behaviour

Wilson (1971) feels that the first step in understanding the behaviour of a given species is a thorough examination of the sensory physiology of the species. Behaviour in social insects is thought to be like a stored program in a biological computer whose activation is a result of certain sensory stimuli. Moser (1970) writes:

> Insects function like tiny robots programmed to do specific jobs. Their nervous systems act like biological computers; they are activated, as if by punch cards, when their receptors are stimulated. The external receptors respond to pressure, sound, light, heat, and chemicals.

The second step involves evaluating their mental capacities in search for peculiarities that may relate to their social achievements. The third step, and most challenging according to Wilson, is explaining collective behaviour as a product of the behaviours of the individual colony members. In this section we will take each step in turn as it applies to social insects.

Sensory Capabilities and Bees. The most studied of all insect species is the honeybee. "Two generations of entomologists have carefully measured the powers of discrimination of the honeybee in every known sensory modality" (Wilson, 1971, p. 197). The main technique used is *Pavlovian* in which the bees are first allowed to feed on sugar while the stimulus to be studied is simultaneously exposed to the bees. Next the bees are tested to see if they are attracted to the stimulus in absence of the reward. If successfully trained in this simple conditioned response they are allowed to choose between the stimulus and an unfamiliar but similar stimulus to see if they can discriminate between the two. Finally, the unfamiliar stimulus is incrementally changed until the smallest detectable difference between it and the stimulus detectable to the insect is determined. This method seems to have reached its limits and newer techniques in the electrophysiological and biochemical fields combined with electron microscopy have revealed new discoveries. Wilson (1971) describes the sensing capabilities of bees:

The composite picture assembled by these behavioral and physiological studies can be roughly grasped by comparing the honeybee's sensory capacities with those of man. The honeybee can see in almost all directions around its body simultaneously, but, compared with man, it is very myopic and receives fuzzy images, even of large, nearby objects. It is not aware of shapes as we appraise them, but it is very sensitive to broken patterns, the flickering of light, and sudden movement. It requires approximately the same amount of light we need to see any image at all. It has color vision, but, instead of the familiar spectrum ranging from blue-violet to red, its sensitivity starts in the ultraviolet and ends in the yellow or the near red. Its ability to sense ultraviolet light allows it to see the sun through an overcast sky on some days when we are unable to accomplish this feat. Moreover, the colors of many flowers and butterfly wings look radically different to it because they bear ultraviolet markings invisible to us, and in a few cases red markings obvious to us but invisible to the bee. The bee can also evaluate the plane of polarization of sunlight, a quality totally alien to our own vision. The bee is virtually deaf to airborne sound but moderately sensitive to groundborne sound, which it detects through its feet. It has a sense of smell almost identical to ours. Its sense of taste is similar, but appears to be generally less sensitive and to have a coarser discriminating power. The bee is very sensitive to touch all over its body, but it apparently has far less capacity for judging the texture of surfaces. It has a comparatively excellent sense of balance and can perform at least one feat beyond our capacity, namely, orientation to gravity at a constant angle while walking up and down a vertical surface. It also possesses at least a limited responsiveness to the earth's magnetic field.

The remarkable ability of the honeybee to detect the plane of polarization of sunlight provides the bee with directional information. It was von Frisch (1949) that determined this unique ability that has subsequently been found in other insects, crustaceans, spiders, mites and even in squids and octopi. When a bee succeeds in finding food it returns to the hive to report this information via the *waggle* dance. The waggle dance is a figure eight pattern performed by the bee on the vertical comb of the nest. The direction of the straight run on the vertical comb and its duration are closely correlated with the direction and distance, respectively, of the food find with reference to the hive (Wilson, 1971). During the straight run the bee vibrates—the waggle—13 to 15 times per second. The angle the straight run makes with the vertical is the same angle with which the bees must orient themselves to the sun in order to find the food. The bee's ability to detect gravity is crucial to the waggle dance.

Heran (1952) determined bees can sense changes in temperature of as little as one-fourth of a degree. This ability is used to maintain precise nest temperature required during the production of honey. Lindauer and Martin (1963) and Martin (1964) proved that odour sensing is achieved by an estimation of differential stimulation of the two antennae. Wilson (1971) describes the experiment:

> Bees were trained to walk up the lower arm of a Y-tube and, upon reaching the fork, to choose the upper arm out of which the appropriate odour emanated. When Martin crossed the antennae of a bee and glued them in place so that the left antennae projected to the right and the right to the left, the bee took the wrong turn. This indicates that the bee

weighs information received from the two antennae concerning the relative concentration of the odorant on either side of the head.

One of the most incredible abilities of honeybees is the ability to keep precise time. This ability becomes even more impressive when witnessed as part of the sun orientation mechanism. Lindauer (1954) and von Frisch (1967) determined bees were able to account for the change in the sun's azimuth using their internal clock. Wilson (1971, p. 209) describes the experiment:

> To take a concrete example, one bee was observed dancing inside a hive in a closed room from 7:05 until 10:46 A.M. in the course of a single morning. During this time the azimuth of the sun shifted 54.5° clockwise, while the bee's direction of dancing shifted 53.5° in the opposite sense. Thus, without any clue whatsoever except its memory of the movement of the sun and its own internal clock, the bee adjusted almost perfectly to the sun's movement and still performed straight runs that pointed symbolically to the target.

Next, we examine the sensory capabilities of ants of which far less is known.

Sensory Capabilities of Ants. The sensory abilities of ants are generally the same as the honeybees. Vision differs among the species from complete blindness to bee-like acuity. Hearing, smell and sense of gravity are comparable with the honeybee. The sense of taste is almost as developed as that of man. No sense to the earth's magnetic field has been found (Wilson, 1971).

Wilson presents some of the evidence behind these statements beginning with vision. Christiane Voss (1967) determined that, although the angle of divergence between the optical axes of adjacent ommatidia is a relatively gross nine to ten degrees, the ants of the *Formica rufa* group can distinguish black and white strips that present visual angles of as little as one-half degree. Voss determined shapes are distinguished in the same way as honeybees.

Sun-compass orientation in ants, and in animals generally, was discovered by the ant taxonomist Felix Santschi (1911). Santschi's interest was piqued by the problem of how workers of desert ants were able to forage at a distance from their nests and then find their way back home over the featureless desert sands, even when strong desert winds made odour trails impractical. He made his discovery by shading the returning ants from the sun on one side and presenting an image of the sun by means of a mirror on the opposite side. The ants reversed direction by 180° and marched away from their nest. When the shade and mirror were removed the ants, once again, reversed direction by 180° and marched home.

Jander (1957) determined that ants were able to keep track of time in the same manner as honeybees and use this ability to keep track of the sun's movement and constantly adjust the angle of their return journeys. Like bees, ants are able to utilize the pattern of polarized light to calculate the position of the sun (Jander, 1957; Vowles, 1950).

Hearing in ants is similar to that in honeybees and consists of receiving groundbourne vibrations perceived by the subgenual organs of the legs (Wilson, 1971). Extreme sensitivity to groundbourne sound has been noted by numerous observers (see Haskins and Enzmann, 1938).

Ants have a gravity receptor system nearly identical to that of the honeybee (Markl, 1962). Like the honeybee, ants can substitute gravity for light signals in maintaining a constant angle. To test this phenomenon, Vowles (1954) first allowed the workers of *Lasius niger* and *Myrmica ruginodis* to run over the surface of a horizontal board while keeping a constant angle to an artificial light. After turning off the light and tilting the board to a vertical position, the ants changed their direction to maintain the same angle—but this time with reference to gravity.

Ants have about as much sense of smell as bees and humans. This generalization is based on the work by Wilson, Bossert and Regnier (1969) and showed that ants of various species are able to detect and move up odour gradients by a lateral movement of both antennae. These results are similar to those found by Martin on honeybees.

Ants sense of taste was investigated by Anneliese Schmidt (1938) and found to be similar to bees but they were much more sensitive to effective compounds. Schmidt was also able to prove that ants, like bees, can sense substances by contact chemoreception with the antennae.

Mental Capacities. The vast array of sensing capabilities in both bees and ants helps explain the interesting and varied behaviour of these social insects. Their mental capacities have been shown as a result of studying the sensory capabilities of social insects.

The training experiments designed to determine the sensory capabilities above, have had the beneficial side effect of allowing researchers to test the capacity of bees to learn in a wide range of environmental circumstances. This learning capacity is impressive in several respects. Wilson (1971) summarizes the experiments as having shown that worker bees are able to learn signals in every known sensor modality. They are quick to learn and can master multiple tasks dependent on several modalities simultaneously. These tasks can be both temporally and spatially ordered as in the programs of visits to different flowers at specific times of the day. Isolated worker bees can be trained to navigate mazes with as many as five turns in sequence in response to such clues as the colour of a spot, the distance between two markers and the angle of a turn in a maze (Kalmus, 1937; Weiss, 1953, 1957). In one experiment, after associating a colour with a reward of 2-molar sucrose solution the bee could remember it for as long as six days. If exposed to the association three times in a row, they can remember the colour for as long as two weeks (Menzel, 1968).

Ants are capable of comparable feats. Workers of *Formica polyctena* can remember their way through mazes for periods up to four days (Chauvin, 1964), while those of *Formica rufa* , operating in a natural environment, can remember four separate landmarks using them in orientation for as long as a week later (Jander, 1957). The incredible ability of both ants and bees to memorize the path and angular velocity of the sun has already been described.

An interesting ability, reported by Jander, is demonstrated during the foraging behaviour of ants. Wilson (1971) has summarized it in the following way:

> Equally impressive is the integrative process that takes place in the brains of bees and ants during foraging trips. The outward bound worker typically winds and loops in tortuous searching patterns until it encounters food. But it then takes a relatively direct route (the "bee-line") in its return trip to the nest. On the basis of his experiments with *Formica* ,Jander suggested that the insect performs a continuous series of calculations analogous to the simplest possible mathematical operation. As it runs outward, according to Jander's interpretation, the ant perceives the constant light source, the sun, and it is aware of the angles it takes relative to that source during each of its twists and turns. For every new direction taken, the product of the angle to the sun times the duration of the outward leg of the run is calculated, and the sum of all these products is divided by the total running time to produce the average (weighted) movement angle to the light. When the insect is

> ready to come home, it need only reverse this mean angle by 180°. The neural machinery for accomplishing such a feat—which in our case would require a compass, a stopwatch, and integral vector calculus—is of course quite unknown.

Wilson remarks that it would be easy to succumb to a sense of wonder and to conclude from these fragments of information that social insects are mentally comparable to vertebrates. However, social insects suffer from severe constraints both in their learning capacities and their innate behaviour patterns to levels far below those attained by higher vertebrates. These constraints have been revealed item by item in the course of empirical research.

Learning is restricted to special conditions and has immediate adaptive value. The skill learned is related to some narrow challenge encountered by the insect during the course of their daily activities. For example, the individual workers ability to memorize the angle of their outward journey relative to the sun while simultaneously accounting for the motion of the sun through the sky. This must be done on each trip out of the nest. If a defective worker could not keep time and memorize the hour, it would lose much of its pollen and nectar crop each day. Ants that could not memorize odours with some precision would soon lose its colony boundaries.

The insects have only a limited ability to transfer memories to assist in the learning of new situations. The severest restriction in insect learning is the absence of the process of transfer learning. An example is *Formica* workers inability to run a mastered maze in reverse, they treat the change as a whole new problem; while rats, in contrast can save time in learning by transfer of the previous information (Schneirla, 1946).

Social insects do not play. Play, as Hinde (1966) has tried to define it biologically, is "a general term for activities which seem to the observer to make no immediate contribution to survival." In mammals, play is displayed most often in the younger individuals and appears to have two functions: first, exploring the environment and social partners; second, perfecting adaptive responses to both. Wilson (1971) claims there is no known "behavior in ants or any other social insects that can be construed as play or social practice behavior approaching the mammalian type."

Social insects exercise a severe economy in communication and response patterns. Fire ants (*Solenopsis saevissima*) use a single trail substance to organize both food retrieval by masses of workers and colony emigration and it is also used in alarm communication (Wilson, 1962). During orientation, as we have seen, two different classes of stimuli, gravitational and visual, can be interchanged without difficulty. Thus, both appear to be used by the same steering mechanism in the brain, and as Wilson notes "this innate limitation has been economically turned to advantage by the honeybees to evolve their waggle dance communication." This same dance is used to recruit workers to all food discoveries, including water, as well as to new nest sites during colony division.

In summary, these limitations of mental capabilities found during experiments on learning performance of honeybees and ants have revealed constraints which hold the potential intelligence of these insects below that of mammals. The frequent use of the same communication signals and responses for two or more very different purposes, provide further limitations on the insect brain (Wilson, 1971, p. 219). Next, we examine the interesting question of collective behaviour.

Collective Behaviour. For behavioural biologists one of the main problems in understanding how an insect society functions is to be able to deduce collective activity from individual behaviour. "Collective behaviour is not simply the sum of each participant's behaviour, as others emerge at the society level" (Pasteels et al. 1987). This they claim creates a paradox—how can individual ants appear so inefficient and disorganized, for example in their nest building activity, while at the same time build highly elaborate nest structures?

To answer such a question, Pasteels et al. feel researchers choose one of several attitudes. The first is to consider that their behaviour is far less random than it appears. The majority of communication or division of labour studies adopt this viewpoint. Ants, of course, are not random particles, they do communicate and subtle forms of division of labour are often observed.

A second attitude is to consider behavioural variance as being irrelevant to the society's functioning. Observations of a society are filtered and descriptions of behavioural sequences are reported in deterministic terms, with only those acts which are deemed functional being reported.

"A radically different attitude is to admit that some randomness at the individual level could be part and parcel of the society's functioning." (Pasteels et al., 1987). Due to their great number, Oster and Wilson (1978) have suggested that social insects can well afford behavioural variance. This variance, they claim, could increase the probability that a social activity will eventually be performed. Their collective reliability more than compensates for the individual inefficiency.

In discussing collective behaviour, Wilson (1971) notes an important first rule concerning mass action; namely, it usually results from conflicting actions of many workers. Individual workers have only a very local perception of the behaviour of nestmates near them, and are largely unaware of the behaviour of the colony as a whole. He sites an example of this phenomena in the process of moving the nest. "As workers stream outward carrying eggs, larvae, and pupae in their mandibles, other workers are busy carrying them back again. Still other workers run back and forth carrying nothing." This same process occurs in the construction of comb cells by honeybees. In order to obtain pieces of wax for cells of their own, the workers regularly tear away walls that are in the process of being constructed by other nestmates (Lindauer, 1952). When viewed at close range these antagonistic actions seem chaotic, however, their final result is almost invariably a well constructed nest.

The emergence of statistical order from competing elements is evident in the marching patterns of army ants. Schneirla (1940) describes the movements of individual workers of the swarm raiding species *Eciton burchelli* as erratic. The swarm moves slowly forward when the ants on the front extend the chemical trail by a small amount before they themselves retreat running back into the swarm. Individually, ants in the swarm are observed to collide with one another. Headon collisions usually result in a changed direction for both ants involved. Workers receiving collisions from the rear usually increase their pace as a result of the collision. "Yet out of all this disorder the characteristic swarm of the *Eciton burchelli* emerges: a roughly elliptical mass of workers, 10-15 m or more across and 1-2 m in depth... with the forward edge growing at a speed of 30 cm a minute." (Wilson, 1971). Schneirla notes the swarm is a result of two antagonistic forces. The first is *pressure* on the individual ant to avoid overcrowding. The second force is *drainage*, which is space vacated by workers subsequently filled by other workers in adjacent crowded areas. The influence of these two forces are felt by wave-like propagations through the swarm.

Wilson (1962) refers to such interaction as "mass communication" and defines it as the transfer among groups of information that a single individual could not pass to another. Wilson explains this mass communication as a result of a behavioural response, in accord with certain probabilities, to the stimuli normally present in the colonial environment. In absence of a stochastic theory of collective behaviour Wilson (1971) proposes a series of widely ranging evolutionary hypothesis:

> (1) the individual social insect , being unaware of most of what is going on in the colony to which it belongs, responds in an ad hoc manner to the stimuli it encounters moment by moment; (2) the responses and the probability of their occurrence are programmed genetically so that mass behavior of the colony is efficient with respect to the particular environmental conditions experienced through evolutionary time by the species; (3) the program evolves as the environment changes, always in the direction of increasing colony efficiency; (4) caste ratios, the age structure of individuals in the colony, and communication also evolve so as to provide the responses and their probability structure with greater efficiency at the colony level.

This view of collective behaviour would allow for the reconstruction of mass behaviour from a knowledge of the behaviour of a single colony member. This view, however, is not shared by all researchers. Pasteels et al. (1987) feel that stochastic events can have a creative role and claim that a "structured but flexible collective activity can emerge from unspecialized workers when non-linear, autocatalytic mechanisms associated with stochastic events regulate their activity." Their approach is inspired from self-organizing processes (Glansdorff and Prigogine, 1971; Nicolis and Prigogine, 1977). By self-organization in insect societies they refer to the emergence of patterns at the level of the society, resulting from interactions between the individuals or between the individuals and the environment. This is different than the self-organization Wilson refers to, in which patterns are due to the execution of evolution crafted pre-determined programs based on age or caste groups.

The best examples of collective behaviour are those involving the cooperative construction of nests. The resulting outcome of the behaviour is very predictable. Researchers can induce building activity by selectively damaging portions of a nest. Some nests are built requiring many workers lifetimes to complete. The existence of such nests leads to the conclusion that the workers interact in an orderly and predictable manner. But how can the workers communicate so effectively over such an extended period of time? (Wilson, 1971). Pierre-Paul Grassé (1959, 1967) suggested that the key process involved "stigmergy" a Greek term meaning "to incite work". Grassé claimed it was the product of work previously completed, rather than direct communication among nestmates that convinced insects to perform additional labour. This theory was based on his observations of nest building by the termites *Cubitermes* and *Macrotermes*. Grassé distinguished three successive stages of construction which are exemplified in the construction of a single foundation arch. Termites begin the first stage in an uncoordinated individual exploration of the environment. The next stage begins with the seemingly random placement of pellets; one pellet placed in one spot by one worker is often picked up and deposited in another spot by a different worker. The final stage involves the seemingly random chance that two or three pellets are placed on top of one another. Termites prefer this structure and continue to place pellets on top forming a column. If no other columns are nearby then work on the column will cease. However, if another column is close the termites will bend the tops of their columns towards one another. Once the two tops meet the arch is complete. The sense which allows the termites to determine the proximity of the neighbouring column has not been discovered but it is hypothesized that it is olfactory (Wilson, 1971, p. 229).

Several researchers challenged Grassé's simple stigmergy explanation. Among them Stuart (1967, 1969) pointed out that Grassé's theory could not account for the shut down when the job is finished. Of course work on an arch would finish when the two arches met. but what stopped the building of new arches? In his own studies on the repair of nest walls by workers Stuart discovered that termites continue to repair the wall until the disturbing stimuli caused by the breach, namely the air currents, are removed. During the repair additional workers are recruited to the scene by odour trails, which stopped once the breach was repaired. Stuart's findings revealed, contrary to Grassé's simple assumptions, that chemical communication is used by termites in the coordination of nest building.

Another remarkable example of collective behaviour is weaver ant nest construction (Wilson and Hölldobler, 1990). These ants construct their nests from green leaves held together by sticky larval silk. In order to construct a wall workers must fold a leaf, they first spread over its surface and randomly tug at any edge they can grasp. One part is turned more easily than the others, and the initial success causes other ants to aid the effort and abandon their own. Like the preference for columns of pellets termites exhibit when constructing arches, weaver ants are able to sense and have a preference towards the turned portion of a leaf.

An interesting paper by Pasteels et al. (1987) compares collective behaviour with self-organizing systems in which both non-linear mechanisms and stochastic events play an essential role. They present a mathematical model which quantifies trail recruitment. Workers are treated as identical units with unspecialized behaviours, largely random and independent of past experience (i.e. no memory). All that is postulated is that the ants are able to lay a trail after feeding, that they have a certain probability of following this trail and that this probability increases with pheromone concentration. Their claim is that this simple model is able to predict the collective behaviour observed in their experiments with actual ants.

Seeley and Levien (1987) approached this same problem, of food source selection, for honeybee societies. The difference in Seeley and Levien's model is that no random component in communication is involved in the emergence of the foraging pattern. Seeley and Levien conclude "that the sophisticated achievements of a colony as a whole can reflect a *small* (their emphasis) set of underlying rules of individual behaviour."

3.3 Summary

In this chapter we have taken a look at an existing example of collective behaviour, the social insects, in the hope that we can determine the salient features of such a society. With their rich variety of sensing capabilities, social insects exhibit a wide range of collective task achieving behaviour which is both productive and efficient in accomplishing a given task. Their repertoire of behaviours is directly related to the large array of sensing capabilities. Collective behaviour, viewed at the society level, seems to be an emergent property of a self organizing system with a few simple rules of interaction. When viewed at the individual insect level, collective behaviour appears to be a collection of random and uncoordinated activity. Chemical communication is used to incite common behaviours which result in a collective behaviour of the group as it propagates throughout the society. Their existence is both encouraging and mystifying as we strive to recreate just such a mechanism to control our artificial populations of task achieving robots.

In the next chapter, we recall Nature's lessons to guide us towards a theory on collective robotic intelligence and propose a model for its implementation.

Chapter 4

Collective Robotic Intelligence:

A Control Theory For Robot Populations

4.1 Introduction—Proposing A Theory

Can intelligent behaviour-based robots achieve tasks collectively? The idea of using intelligent agents that cooperate to achieve tasks without explicit coordination or communication is not new. Brooks proposes sending a colony of small robots to the Moon to construct a permanent base (Brooks and Flynn, 1989). Dario et al. proposes a social organization of *societies* of cellular mobile robots where useful tasks are carried out through collaboration rather than individual effort (Dario et al., 1991). Additionally, Yuta and Premvuti describe an approach to cooperation of multiple autonomous mobile robots using environmental resources while working toward a common goal (Yuta and Premvuti, 1991).

In this chapter we propose the first steps towards a control theory for robot populations. We begin by first reviewing the lessons learned from the several examples of collective behaviour found in the social insects. Next, we present our theory on Collective Robotic Intelligence emphasizing its emergent property. Finally, we present its model of group control and illustrate its use by considering the steps involve[illegible] designing a collective task.

4.2 Towards a Theory on Collective Intelligence

A number of interesting observations can be made from the examples of collective behaviour in the preceding chapter.

1. Social insects have a large array of sensors allowing behavioural components to interact with the environment directly. It is this large array of sensors that seem to allow the insects to display a variety of different behaviours. A behaviour is usually produced as a result of the insect sensing the necessary stimulus. For example, the three ocelli, used for detecting light intensity, governs the amount of time used for food collecting behaviours. Bees with their ocelli blinded still continue to collect food but on a reduced time schedule (Schricker, 1965). Similarly, swarming behaviour in ants can be triggered upon the detection of the light of dawn (Schneirla, 1956). This simple mechanism of invoking behaviours allows the externally sensed environment to provide the synchronization necessary for temporally ordered tasks. The perceived behavioural complexity is directly related to both the number of sensor modalities and the sensor's individual sensitivity.

2. Collective behaviour usually begins with seemingly random and uncoordinated activity. The activity usually results in a form of self-organization. The resulting outcome of the behaviour is very predictable. For example, arch construction by the termites *Cubitermes* and *Macrotermes* begins with the random placement of pellets,

some of which end up piled on top of another pellet. This activity results in the formation of arch columns, a structure the termites have a preference for, and they continue to pile pellets on top (Pierre-Paul Grassé, 1959, 1967). Weaver ants begin leaf folding (part of their nest building activity) by grabbing and pulling any corner they can grasp. This eventually folds the leaf at a weak corner, ants will then abandon their corner for the newly folded one (Wilson and Hölldobler, 1990).

3. Collective behaviour can be viewed as an emergent property of the self-organizing system with a few simple rules of interaction. This emergent property results from the system interacting with its dynamic environment. The specification of individual behaviour alone does not explain the resulting functionality of the group. The rules of interaction and the environment in which the interaction takes place must be accounted for together. The implication when designing collective tasks is that an interaction loop must be found comprising of the system and the environment which ultimately converges towards the desired goal. For example, the elliptical swarm pattern that emerges from the army ants *Eciton burchelli,* mentioned in the previous chapter, resulted from the two simple antagonistic rules of *drainage* and *pressure.* Pasteels et al. (1987) were able to reproduce in simulation the foraging patterns of the ants *Tetramorium caespitum* by two simple rules of behaviour, namely, lay trail after feeding and follow trail scent. The environment is used, in these examples, as a communication medium. These simple behaviours were able to generate the coherent collective behaviour observed in field experiments.

4. Subtle forms of communication take place in collective activity. Most of this communication is chemical based and used to trigger a given behaviour. For example, in Stuart's (1967, 1969) nest reparation experiments, termites used chemical odour to attract additional workers to the site of repair activity. Simple, non-chemical

communication is also used to incite behaviour. For example, tandem running is believed to be the predecessor to odour trail systems (Wilson, 1971). Tandem running consists of a leader and a follower ant. The leader ant stops running in order for the follower to bump and push the leader forward; the leader continues for another 3 to 10 centimeters at which time it stops again.

5. There is evidence that suggests that ants alter their behaviour when found in large groups. For example, worker ants were found to excavate the soil and attend larvae at a higher rate when found in large groups (Hangartner, 1969). Another example is the aggressiveness of an individual ant increases as the size of the crowd of nestmates around her grows (Wilson and Hölldobler, 1990). Wilson (1971) found that when workers of *Acanthomyops claviger* are kept in solitude, they are nearly insensitive to the natural alarm substances of the species. In contrast, those placed in the same nests with a few hundred nestmates respond normally to the alarm substances. It has been theorized that some of these group behaviours are a result of the sensing of higher concentrations of metabolic products associated with large numbers of ants.

6. Can ants communicate with themselves? Foraging workers were found to do just that when they dispense orientation pheromones in their odour trails and then follow the traces during the return journey to the target areas (Hölldobler et al., 1974).

7. The effects of collective behaviour form a kind of distributed representation. Take for example, nest construction in which the nest is physically realized as a result of hundreds of tiny pieces put together by individual ants. The object "nest" is spatially distributed with each ant holding a partial representation of the whole.

8. Task behaviour can be considered temporally ordered like a Finite State Machine (FSM). Consider the task of nest construction. The task involves a series of steps that must be performed one after the other. If we were to consider each step a state of a

FSM then the system would move from state to state. The transition between states occurs when the agent (in this case an ant) senses a change in the environment. For example, termites begin nest building by exploring their environment (state 1). They start by picking up pellets and moving them about (state 2). Once they sense two pellets on top of one another (a random event) they begin to pile other pellets on top of the structure (state 3). The structure continues to grow until another column is sensed nearby, at which point the termites begin to bend their columns towards each other forming an arch (state 4). If no other column is sensed nearby they abandon their work and begin moving pellets around again (state 2).

Can simple autonomous robots achieve tasks collectively? Considering the examples mentioned of collective task achieving behaviour by ants we are tempted to answer yes. Ants can be compared to simple robots executing equally simple programs. There is no evidence to suggest that any one ant understands the resulting outcome of its behaviour on the whole; no master architect directing hundreds of workers, but rather, stored behavioural programs that can be invoked by researchers using appropriate stimuli. For example, corpse removal is another collective behaviour invoked by chemical odour. Dead ants are carried away from the nest and discarded in refuse piles by workers. Wilson et al. (1958) were able to invoke the same behaviour from ants by treating bits of paper with acetone extracts of dead corpses. In fact, live ants were also dragged away by nestmates towards refuse piles when small amounts of acetone extract was daubed on them.

These observations on collective behaviour create a new way to view the representation of a collective task as a set of temporally ordered behaviours. This representation is distributed and manifests itself as a sequence of steps necessary to physically execute the individual components of the task. Take for example nest building, each individual has a sequential program used to build pieces that will eventually form a physical nest. Each step in the program is a side effect of a behaviour. The behaviour becomes active when the appropriate stimuli exist in the environment. Therefore, a physical piece of the nest can be described in terms of a set of sequential behaviours (steps in the program). The behaviours are temporally ordered in a cascading fashion (i.e. they must follow one another to create the piece, with each step automatically following the last). The representation is distributed because the nest is really a sum of each of these pieces, and can only exist once the entire collection of individuals are present.

This type of intelligence is emergent because all the individual behavioural bits of representation need to be together and operate simultaneously in order to achieve the collective task. Emergent functionality describes a function that is not achieved directly by a behaviour, but indirectly by the dynamic interaction of more primitive behaviours among themselves and with the world (Steels, 1991). Emergent functionality has become one of the main themes in research on Artificial Life.

Artificial Life (AL) is a bottom-up approach to constructing man-made systems that exhibit behaviours characteristic of natural living systems. It views an organism as a large population of *simple* machines, and works upwards *synthetically* from there—constructing large aggregates of simple, rule-governed objects which interact with one another nonlinearly in the support of life-like, global dynamics. Emergent behaviour has become the key concept in AL. It is this bottom-up, distributed, local determination of behaviour that AL employs in its primary methodological approach to the generation of life-like behaviours (Langton, 1988).

From the previous observations we propose the following control theory: *Collective Robotic Intelligence (CRI) is an emergent property of a collective task; it is represented* *as a distributed cascade of sequential behaviours and the population control is achieved by invoking group behaviour.* CRI states that control of a population of robots is achieved by invoking group behaviour. Several different mechanisms may be used to invoke the necessary group behaviour outlined below in the following lemmas:

1. Group behaviour may result by using a common goal oriented collective task. For example, if we design a group of robots to locate and converge upon a single object and while doing so the robots do not interfere with one another, we have then successfully controlled the movement of the group as a whole. This type of group control can be observed in ants that collectively move a large object towards their nest opening.

2. Group behaviour may result by herding robots. This may be accomplished if we design our robots to stay together in "herds" using a FOLLOW behaviour. The collective task may require that groups of robots remain together. Consider grass cutting by a group of small robots who travel in herds. The control of the group is accomplished using a behaviour designed to keep them together: the task is accomplished by having all the robots execute a function (i.e. cutting) while the group is moved through its environment. This mechanism is used by ants in recruitment such as tandem-running, where ants will recruit another to follow or in swarming behaviour where groups fan-out by following odour trails.

3. Group behaviour may result from environmental cues. This method of control uses cues found in the environment to invoke the same behaviour in the group. Both dawn and dusk provide ants with a visual cue to begin or end food collecting behaviours and are an example of an environmental cue. Collective tasks can be designed to allow the environment, in which the robots work, to provide the cue that invokes the group behaviour. For example, a group of building cleaning robots,

designed to keep the outside surface of buildings clean, would begin their activity at dusk by depositing photosensitive chemicals on the outside surface. Cleaning action would begin the next day when the chemical reacted with sunlight.

4. Group behaviour may be invoked once the robot becomes aware it is in a group. This method of control is accomplished by equipping each robot with sensors to detect the presence of other robots. Sensors would be placed around the periphery of the robot enabling it to detect other robots both in front and behind as well as to the left and right. The collective task is then executed once the robot finds itself surrounded by other robots. For example, bulldoser robots designed to level the ground would only be effective once a large group was formed travelling in the same direction. The robots in the centre of the group would execute the collective task while those on the periphery would be responsible for navigating the group over the surface to be leveled. This form of control is found in ants that exhibit certain behaviour when found in groups only (see observation 5 above for several examples).

5. Group behaviour may also be invoked through autostimulation. The wide range of alarm substances found in ants is an example of this control mechanism. For example, if we are using a group of robots to search an area for a particular substance, then once a single robots finds the substance it broadcasts a signal which invokes a behaviour in all robots receiving the signal. This method of control is different from the above methods because it is a form of self-facilitation (a term taken from the psychological literature meaning communication that promotes rather than inhibits activity).

In the next section we propose a model of individual control and outline the necessary features that allow the model to be used in the collective control of a population of autonomous robots.

4.3 A Model of Group Control

In order to test any theory on collective behaviour in a meaningful way, we will need to build a group of robots based on some individual model of control. This collection of robots should then form a system with emergent functionality. Steels (1991) defines the characteristics of such a system with emergent functionality as consisting of (1) a set of behaviours each with its own interaction with the environment; (2) the activation of a behaviour is a direct result of stimulus from the environment and; (3) behaviours contribute individually to the global functionality but, require other behaviours to do useful work. In a system with emergent functionality all the components need to be present and operate simultaneously to achieve the desired collective task.

Our model based on these principles is shown below in figure 6. The intent is not to model ants, but rather to use the lessons we have learned about collective behaviour to propose a useful control model for the individual robot. The model and its behaviours are designed to allow a group of robots to interact. Control of the group is achieved using any one of the five methods outlined in the previous section. Using this model we will outline the steps in designing a robot to be used in a collective task.

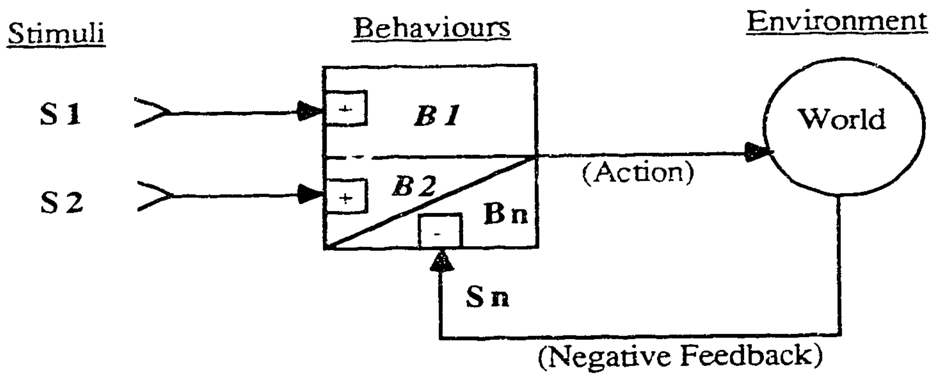

Figure 6: Control model of an individual robot. Behaviours *B1* and *B2* both receive a positive stimulus, whereas behaviour *Bn* receives a negative feedback stimulus.

The above model is best explained by defining the sequence of steps involved in designing a collective robotic system. As an example, we will choose the simple task of locating and pushing a box collectively; this task uses the first method of group control, a common task, outlined in the previous section. Cooperation is achieved among the robots by not interfering with others in the group. The task is such that it can not be accomplished with a single robot (i.e. the box is too heavy or turns on its axis). The steps are as follows:

1. *Define the collective task as a sequence of behaviours.* These behaviours may either be reflexive (i.e. without memory) or reactive (i.e. with memory). Each behaviour interacts directly with the environment, receiving immediate feedback. A behaviour becomes active when its associated sensor detects the preconditions necessary for activation. For example, the box pushing task will require an *explore* behaviour to allow the robot to explore its environment; a *locate-box* behaviour to identify the box; and an *avoid-obstacle* behaviour for collision avoidance. The sequence would simply consist of *explore* then *locate-box*. The explore behaviour would allow the robot to search for boxes; the locate-box behaviour would take control

once a box is sensed and guide the robot towards the box. If on the way an obstacle is encountered, the avoid-obstacle behaviour would cause the robot to veer from its path.

2. *Determine the stimulus necessary to activate each behaviour.* For example, the explore behaviour is active if the box sensors are off; the locate-box behaviour is active if the box sensors are on; the avoid-obstacle behaviour is active if any one of many obstacle sensors are on.

3. *Determine the minimum number of robots needed to accomplish the task.* This is task dependent and in the case of our box pushing robots at least two robots are required.

4. *Determine the stimulus which causes negative feedback.* This is required in order for the robots to determine if progress is being made toward the collective task. When using reflexive behaviours only, this feedback will allow for the detection of cyclic behaviour (see chapter 2). In the our case, should a robot find itself pushing on a side opposite three other robots, this stimulus and its associated behaviour will allow the robot to stop pushing and retreat (see figure 7).

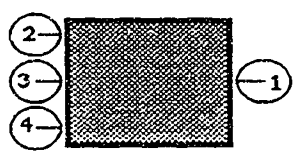

Figure 7: Negative feedback will allow robot number 1 to evaluate its progress.

Consider another example of the use of multiple robots. The following example illustrates a group task which uses CRI. The group task we wish to accomplish is that of *bulldozing* small bits of styrofoam evenly spread throughout an area into a pile at one end of the area. In order to accomplish this task, it will be necessary for the collection of robots to form a physical configuration and move as a unified group. This configuration may be a square grid, a horizontal line, or possibly a geese-like "V" formation, exhibited when geese flock, and would be task specific. Group task behaviours, such as bulldozing, emerge and are a result of individual behaviours invoked by the robots, as they become aware of the group.

Each robot is equipped with several task achieving behaviours designed for the particular group task at hand. A brief description of each behaviour follows:

FIND	Causes the robot to search for similar robots.
AVOIDR	Causes the robot to avoid obstacles on the right.
AVOIDL	Causes the robot to avoid obstacles on the left.
FOLLOW	Causes the robot to follow another robot.
PLOW	Causes the robot to lower its plow.

Each robot is equipped with infrared sensors for detecting other robots: in front; behind; to the left; and to the right. Robots are capable of detecting and following other similar robots. This results in the robots forming and travelling in fixed physical configurations as illustrated in the diagram below:

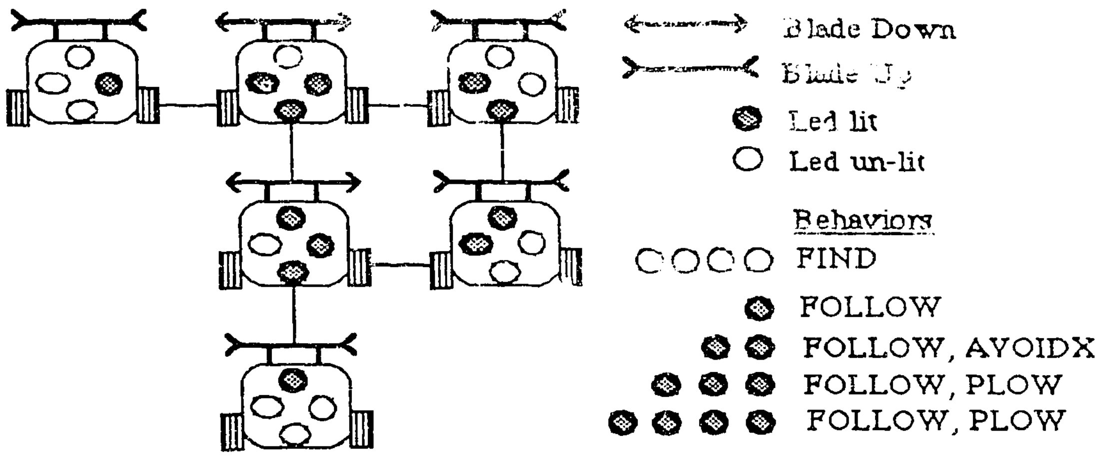

Figure 8: A group of plowing robots.

Initially, the collection of robots exist unaware of each other. Guided by their FIND behaviour the robot attempts to seek and identify similar robots. Once a similar robot is found, indicated by a single lit LED (Light Emitting Diode), the FOLLOW behaviour guides the robot to follow. As other robots join the group, those located on the group's periphery become responsible for collision avoidance on the side without a fellow robot. This is accomplished with the *avoid-left* (AVOIDL) and *avoid-right* (AVOIDR) behaviors. The PLOW behaviour is invoked once the robot is aware of other robots on at least three of its four sides.

The group bulldozing behaviour is a result of a sufficient number of individual robots collectively plowing. The resulting level of competence, exhibited by the group, is higher than that of any individual robot, and is greater than the sum of the individual robot capabilities. To see that, consider a single robot trying to push a spherical ball. Each time the robot attempts to push the ball in a forward direction, the ball rolls off to one side of the robot. In order to move the ball forward two robots will be required pushing collectively in the same direction.

Where would we use such collections of multiple robots? Consider the problem, previously mentioned, of keeping the outside surface of buildings clean We could design a surface clinging robot. Swarms of these surface clinging robots deposit chemicals to sites unreachable by other means. Behaviours would be designed so that these robots stay hidden during the daylight hours only to come out and deposit their light sensitive chemicals at night. Cleaning action, by the chemical, would begin the next day as the chemical reacts with daylight.

4.4 Summary

Based on the observations of collective behaviour from the last chapter, we have taken the first steps towards a theory on collective robotic intelligence and proposed a new way of representing collective tasks as a distributed sequence of behaviours. The resulting intelligence is emergent because all the individual behavioural bits of representation need to be together and operate simultaneously in order to achieve the collective task. Control of the population of robots is achieved by invoking group behaviours. Five methods for invoking group behaviour are presented with several examples given. Finally, we propose a control model used in the design of these collective systems and demonstrate its use by considering an example of a collective task. We speculate that populations of these simple robots can be designed to perform useful real-world tasks effectively.

In the next chapter, we present our robot population simulator, *SimbotCity*, and describe its architecture and facilities for testing the control model and the dynamic interaction of multiple robots.

Chapter 5

SimbotCity:

A Robot Population Simulator

5.1 Introduction—Testing the Theory

The ultimate goal of this work is to design and build a number of real physical robots capable of achieving simple tasks collectively. There are two approaches we may adopt in order to test the control model developed in the last chapter. First, we can design then construct several identical physical robots, testing their performance and dynamic interaction in the real world. This could potentially be very time consuming, as any change in the control model would have to be replicated in all the robots. The second approach is to test the control model in simulation before constructing the physical robots. The amount of time needed to change and test a new control model would be reduced, as would the error in replicating the model in each simulated robot. The disadvantage with simulation stems from the simplifications made in modelling any aspect of the real world. However, this disadvantage is minimized if we treat the simulation as a tool for testing the feasibility of our control strategy; realizing the ultimate test still lies in the real world.

In this chapter we present our robot population simulator *SimbotCity* and discuss its use as a tool for investigating control strategies used to control populations of mobile robots. We begin by outlining our simulation objectives by defining some of the new issues that arise in robot populations. Next, we present SimbotCity's architecture which includes models for sensors, behaviours and actuators; these models are then combined using a control architecture explained in the following section. Behaviour implementation is discussed next, using the *push-box* task and its behaviours as an example. A collective task and its experiments are then presented. Finally, related work in simulation from the fields of Graphics and Simulation of Adaptive Behaviour is discussed.

5.2 Simulation Objectives

When multiple robots start to interact a whole series of new issues begin to surface. Brooks (1991) outlined several of these issues, a subset of which we shall consider here.

Emergence: Each robot's control system consists of a set of behaviours, we would like to see what the collective behaviour of a group of homogeneous robots will be. Further, if an incremental modification to the individual robot is made, we would like to determine its effect on the collective behaviour of the group.

Cooperation: In achieving collective tasks, some form of cooperation will be necessary. This may simply take the form of not interfering with other robots as Yuta and Premvuti (1991) have proposed or may involve some other form of cooperation.

Allocation: We would like to know the minimum number of robots necessary to accomplish the collective task. Brooks refers to this as "density dependence." Also, should we decide to use more than the minimum number, at what point does the system cease to be functional due to a glut of robots?

Herding: There are advantages in keeping a group of robots together. Collectively they can respond much quicker to a given stimulus than if they are more spatially distributed. An example might be a group of fire fighting robots whose *extinguish* behaviour activates upon fire detection. A group would respond quicker to the blaze and gain control easier than just one robot. Given the local perceptive abilities of the robots, what are the suitable behaviours needed for herding?

These are the issues we wish to investigate with simulation; and the lessons learned in the process, will serve as a guide when building the physical system.

5.3 The Architecture of SimbotCity

The simulator's architecture has been implemented in a modular fashion making extensive use of abstract data types (ADTs). A population model consists of a group of robots. Each robot consists of a *sensor* model, a *behaviour* model and an *actuator* model. Models are further subdivided into model *types*. For example, a robot may have several different sensor types: infrared for close range obstacle detection; sonar for long range obstacle detection; and acoustic for sound detection to name a few.

Similarly, behaviour and actuator models are also divided into types. Models are implemented as ADTs and are accessed using the appropriate function. So as not to sacrifice speed of execution, functions are implemented as *macros*.

The simulator is implemented on a SUN Sparc station and runs under the X-Window system. User interaction is designed using the XView[1] graphical user interface which is an implementation of the OPENLOOK specification. The current version of SimbotCity, version 1.08, consists of approximately 2000 lines of C code.

The user interface of the simulator is illustrated in figure 9. A scrollable window allows the user to view a portion of the robots' environment. To create a population of robots, the user creates a configuration file. The configuration file contains an entry for each robot consisting of the robot's number; initial X and Y coordinate; and direction the robot is pointing. Configurations are then dynamically loaded into the simulator at run time. A simulation may be run continuously or single stepped. Single-stepping is useful in debug mode in which each robot's behavioural parameters are displayed. User interaction is handled with panel buttons eliminating the need for command memorization.

---

[1]XView Programming Manual for Version 11 of the X-Window System by Dan Heller, O'Reilly & Associates, Inc. 1990.

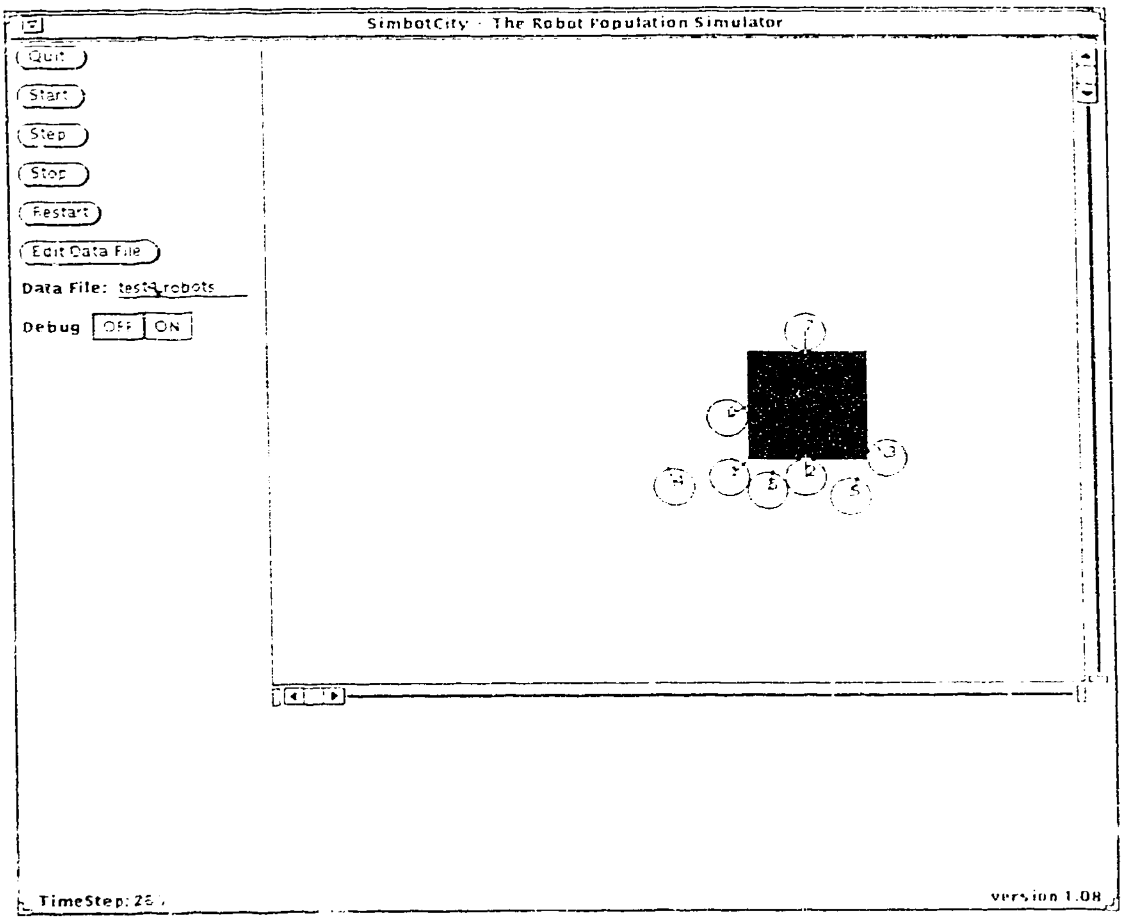

Figure 9: The user interface of SimbotCity version 1.08.

5.3.1 The Robot Model

In the current version of SimbotCity each robot has three sensors: a goal sensor (S2), an obstacle sensor (S1), a robot sensor (S0), and two actuators: a left wheel motor (A1) and a right wheel motor (A0). Each robot's control model consists of five behaviours: a *goal* behaviour which directs the robot towards the goal, an *avoid* behaviour which steers the robot clear of obstacles including other robots, a *follow* behaviour which allows one robot to follow another, a *slow* behaviour which prevents rear end collisions between following robots and a *find* behaviour which causes the robot to explore its environment. A direction point . indicates the robot's current orientation as illustrated below in figure 10:

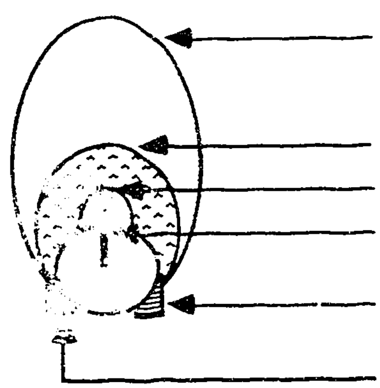

Figure 10: An example of one instance of the robot model with three sensors and two actuators.

5.3.2 The Sensor Model

The sensor model is implemented as an ADT with six pieces of information: sensor number, type, direction sensor is pointing, view angle width, input value for active sensors, and output value.

```
struct sensor {
        int number;
        int type;
        float direction;
        float view_angle;
        float input_value;
        float output_value;
};
```

Macros provide access to the data structure and sensor processing is based on sensor type.

Currently there are five sensor types each based on the available physical sensors in our lab. These types are: infrared, sonar, acoustic, light, and switch. In SimbotCity sensor types may only represent the actual physical sensors available. This approach ensures that simulated robots can eventually be built using the same sensin techniques.

5.3.3 The Actuator Model

The actuator model is implemented as an ADT with five pieces of information: actuator number, type, position on the robot, input value and an on-off switch.

```
struct actuator{
        int number;
        int type;
        int position;
        int input_value;
        int onswitch;
};
```

Macros provide access t the data structure and actuator processing is based on actuator type. There are four types of actuators: motor, hand, plow, and solenoid. The current robots use the motor type only; a left and right motor provide movement. Steering the robot is accomplished by turning one motor on at a tim For exampl   steer the robot left the right motor is turned on while the left motor is turned off.

5.3.4 The Behaviour Model

The behaviour model is implemented as a mapping between sen   inputs and actuator outputs (i.e. reflexive behaviour). Sensors provide the input to behaviours, which then process the sensor data and output the results to the actuators. Each behaviour reads its connected sensors and calculates a given response during each simulation time step. The resulting command is sent to the actuators to be used during actuator processing. The behaviours are arranged in a fixed priority with the highest priority behaviour sending its actuator commands last.

5.4 Control Architecture

The control system we have developed for our robots is modelled after Jon Connell's modified subsumption architecture (Connell, 1990). An example of the architecture for the control system of our robots is shown in figure 11. It consists of a number of modules (Bn's), each of which implements one behaviour. The modules receive input from the sensors (Sn's) and generates output commands to the actuators (An's). Behaviour output commands are combined in a fixed priority scheme (circles) with only one module at a time controlling the actuator. Connell's architecture has no direct channels between modules and no central communications. Two special constructs determine how the behaviour modules interact. First, a module can inhibit the input or output of another module (circle with an "I") thereby activating or disabling the behaviour module. This is a slight modification from Connell's original proposal (he does not allow inhibition of inputs). Second, a module can suppress the output of another module (circle with an "S") by replacing the output with its own output commands.

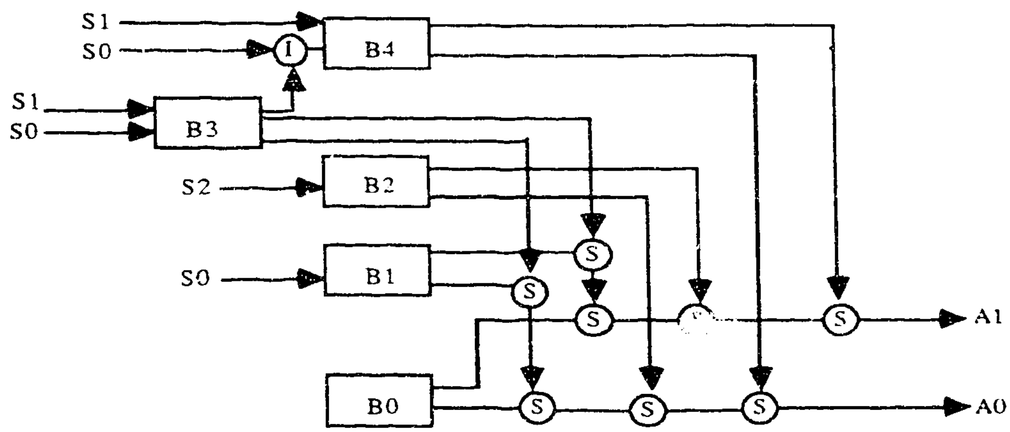

Figure 11: The control system consists of behaviour modules (the B's), each of which processes the sensor data (the S's) and generates commands to the actuators (the A's). Behaviour arbitration is through a fixed priority network (the circles).

The behaviours in figure 11 for our simulated robots are: FIND (B0), a behaviour that causes the robot to move in a spiral pattern used to explore the environment; FOLLOW (B1), which uses the S0 robot sensor to detect other robots and causes the robot to follow it; GOAL (B2), which uses the S2 goal sensor to locate the box and steer the robot towards it; SLOW (B3), which reads the S0 robot sensor and the S1 obstacle sensor to slow the robot down preventing a collision with another robot (both S0 and S1 are on); AVOID (B4), this behaviour uses the S0 and S1 sensors to detect an obstacle (S0 = off, S1 = on) and steers the robot away as long as the sensors are active. This behaviour can also be invoked by the SLOW behaviour if SLOW can not prevent a robot to robot collision; this is accomplished by inhibiting the S0 sensor which generates the same obstacle detection pattern (S0 = off and S1 = on).

Behaviour arbitration is handled by the fixed priority network indicated in figure 11 by circles. The FOLLOW behaviour can suppress the FIND behaviour's motor commands and replace them with its own motor commands. This occurs when other robots are detected in the vicinity (discussed further in the section on behaviour implementation). Likewise, the SLOW behaviour can suppress the FOLLOW behaviour's motor commands in the event a robot gets too close to the one it is following. Both the GOAL and AVOID behaviours can inject their own motor commands suppressing all others, with the AVOID behaviour having higher priority.

Behaviour modules are designed by deciding the sensory input that activates the module and the action the module will take once activated. Modules are then combined using the inhibit and suppress constructs mentioned above.

5.5 Behaviour Implementation

As an example, SimbotCity (1.08) can be designed to accomplish the collective task of box-pushing. The task is such that it cannot be accomplished by one robot because the box is too heavy to push; therefore, the collective effort of several robots is necessary. To accomplish this task the robots must locate the box, move towards it while avoiding collisions with other robots, distribute themselves along a side of the box and push. As discussed in section 5.2, there is an advantage in keeping robots together in a herd when accomplishing collective tasks. Robots are designed to travel in groups by following other robots. The following sections, with their accompanying figures, describe the behaviours necessary to accomplish the collective task of box-pushing.

5.5.1 The FIND Behaviour

The FIND behaviour is the robot's default behaviour and causes the robot to execute a wide spiralling search pattern to the right. Robots move forward only and therefore are in constant motion in their environment. The size of the spiral is governed by a *turn_rate* parameter. The behaviour is constantly active issuing motor commands and requires no sensor input. If no other behaviour sends motor commands then FIND guides the robot.

5.5.2 The FOLLOW Behaviour

The follow behaviour is designed to keep robots together in a herd. The follow behaviour becomes active when another robot is detected by the S0 robot sensor and causes the robot to move towards it. Sensor S0 is a forward looking sensor with a view angle of 90°, thus, robots only follow other robots ahead of themselves. Once a robot is following another, it narrows its viewangle to 22° so as not to be distracted by other groups [illegible]s passing in close proximity. Modifying a behaviour's response in this fashion, is what we refer to as a *behaviour preference*. Behaviour preferences are a technique that allow for the robot to dynamically adapt its response to the environment given the current state of the robot. Adaptive behaviour is a method taken by the *Animat* approach (for an example see Meyer and Guillot, 1991; Wilson, 1991). An example of the FOLLOW behaviour is illustrated in figure 12 taken at TimeStep 134. Figure 13, taken at TimeStep 171, shows the resulting collisions between robots with only the FIND and FOLLOW behaviours active (robot number 2 collides with number 7).

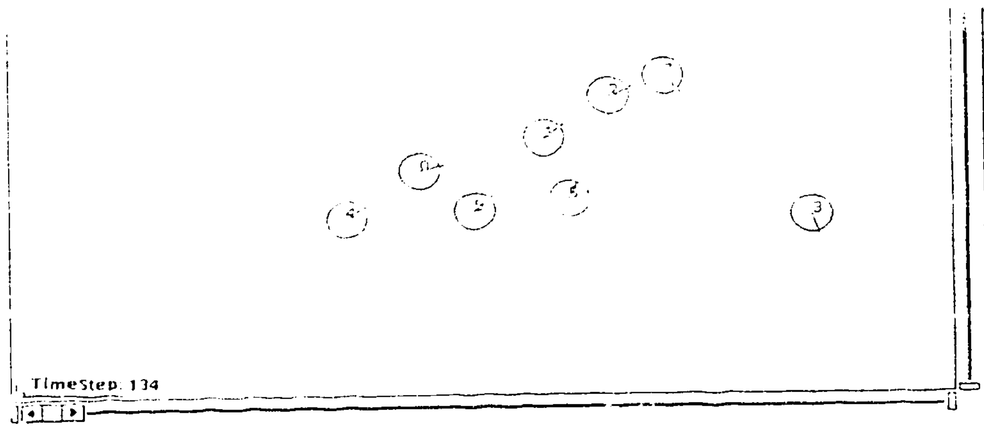

Figure 12: Robots with the FIND and FOLLOW behaviours.

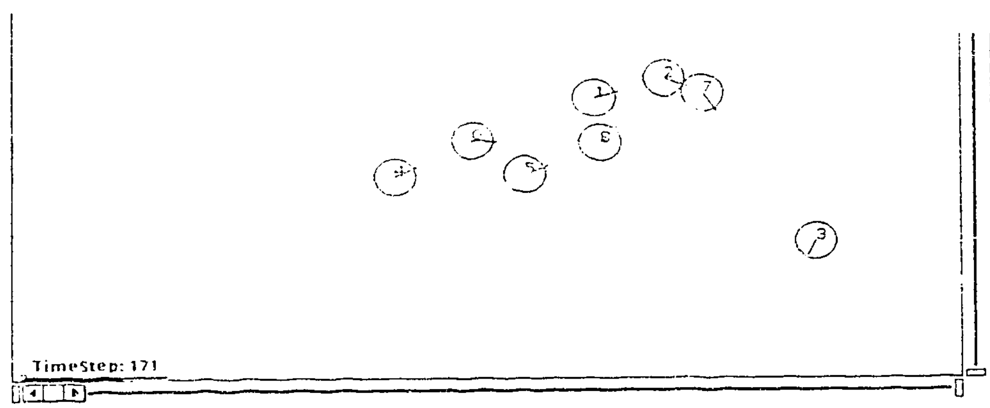

Figure 13: Robots 2 and 7 collide due to no obstacle avoidance behaviour.

5.5.3 The SLOW Behaviour

The SLOW behaviour is designed to prevent collisions between robots following one another. The behaviour becomes active when the S0 robot sensor and S1 obstacle sensor both become active. The behaviour turns both wheel motors off for one Time Step. This effectively slows the robot down and prevents the collision as illustrated in figure 14 taken at Time Step 134, but with the SLOW behaviour added to the control architecture. The distance between robot number 2 and 7 is greater than in figure 12 in which robots have only the FIND and FOLLOW behaviours active.

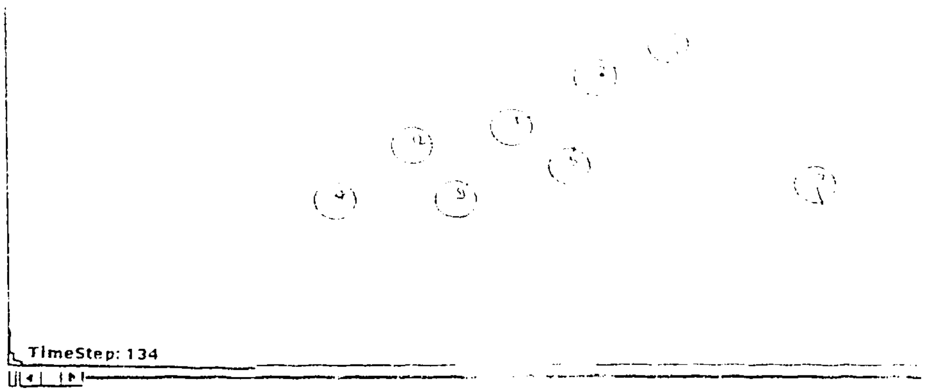

Figure 14: Robots with the SLOW behaviour added.

5.5.4 The AVOID Behaviour

The AVOID behaviour steers the robot away from obstacles. The behaviour becomes active when the S0 robot sensor is off and the S1 obstacle sensor is on. Once active the behaviour commands the left motor on and right motor off for one Time Step resulting in a right turn. Turning continues as long as the sensor pattern remains. The AVOID behaviour can also be invoked by the SLOW behaviour.by inhibiting the S0 sensor input. This occurs when the SLOW behaviour has not been successful in preventing a robot to robot collision. Figure 15 is taken at Time Step 283 just before robots 1 and 2 invoke their AVOID behaviour. Figure 16, taken at Time Step 321 shows the results of the AVOID behaviour between robots 1 and 2. The AVOID behaviour receives negative feedback from the environment allowing the robot to gauge the progress it is making moving forward. For example, in Figure 17 robot 7 is pushing the box in the opposite direction of robots 1 and 2; should the box end up moving against robot 7, pushing it backwards, its negative feedback will cause it to turn away until it is no longer moving backwards as illustrated in Figure 18 taken at Time Step 318.

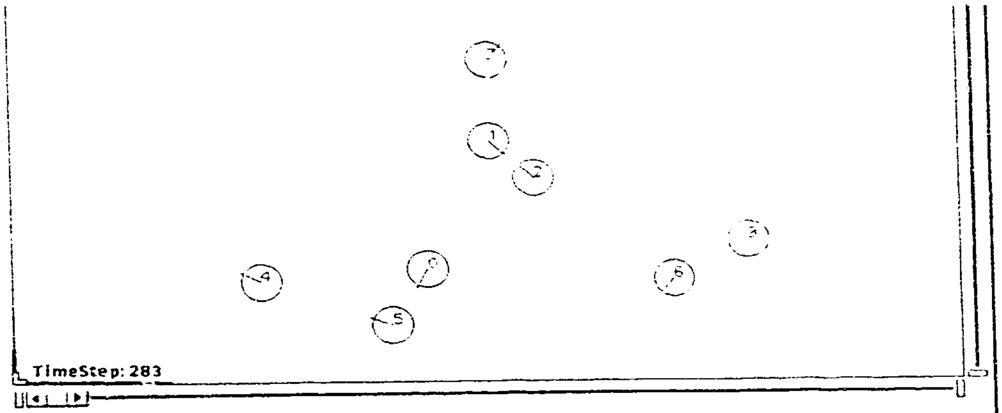

Figure 15: Robots 1 and 2 before invoking AVOID behaviour.

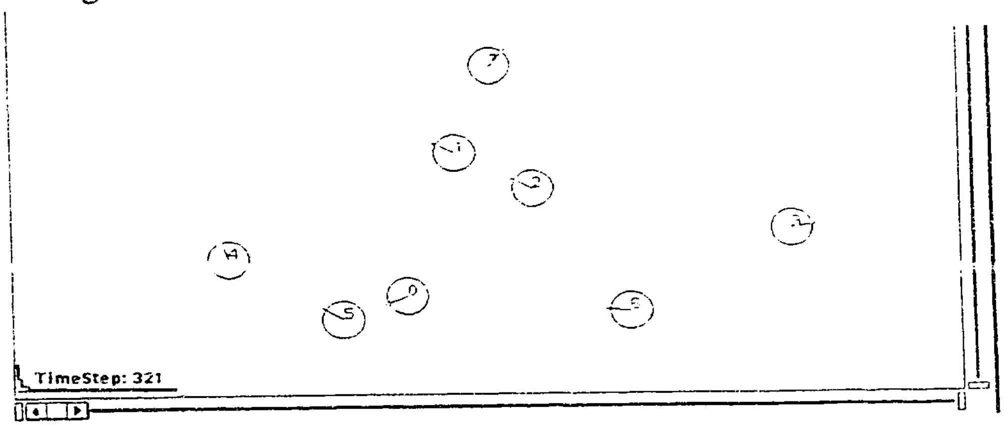

Figure 16: Robots 1 and 2 after invoking the AVOID behaviour.

5.4.5 The GOAL Behaviour

The GOAL behaviour causes the robots to move towards the goal, in this case a box (see figure 17). The behaviour is activated when the robots sense the goal with their S2 goal sensor. This causes the robots to move towards the goal and is proportional to the distance from the goal. The closer the robot is to the goal the greater its desire to move towards the goal instead of towards other robots.

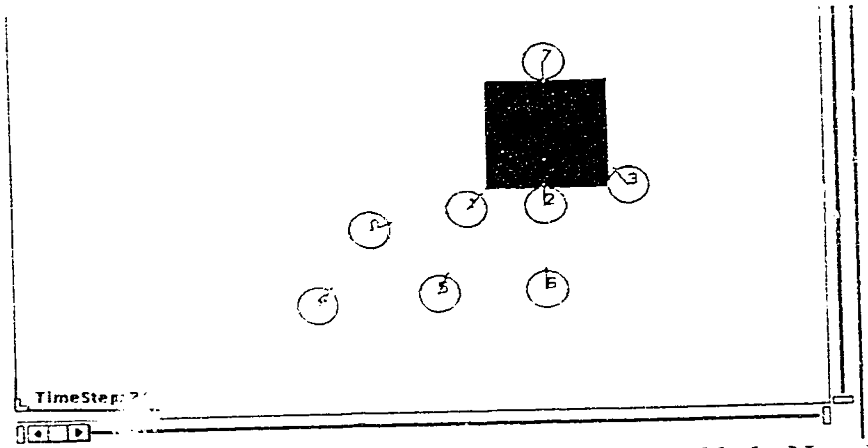

Figure 17: Robots with the GOAL behaviour added. Note Robot 7 is not making progress.

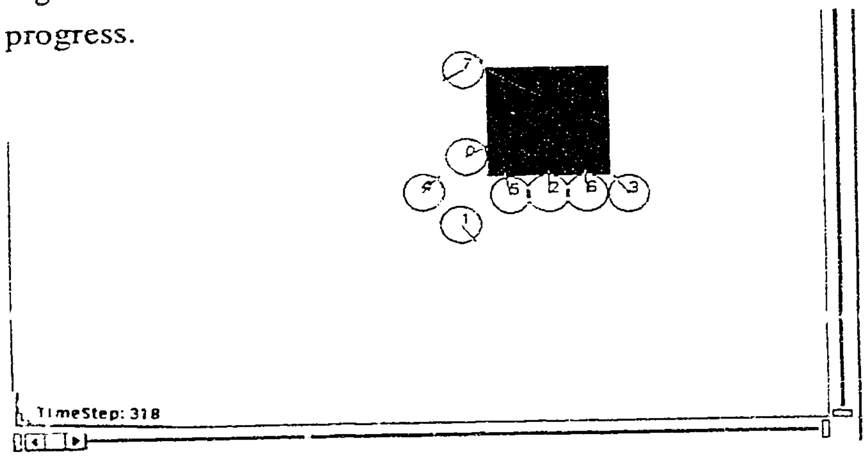

Figure 18: Robot 7 changes direction due to negative feedback connection on its AVOID behaviour.

Together the five behaviours implement the control model necessary to achieve the collective box-pushing task. Robots roam their environment in herds looking for boxes to push. Once found, the robots push the box to one edge of their world. Robots employ a simple form of cooperation by avoiding and therefore not interfering with each other while performing the task. For box-pushing, at least two robots are required to accomplish the task and greater than twelve causes the system of robots to be inefficient due to the even distribution of robots around the perimeter of the box. By keeping robots together in a herd the system of robots responds quicker to the task at hand due to the simultaneous sensing of the box by several robots. This increases the likelihood that the distribution of robots around the perimeter of the box will be asymmetrical causing the box to move in one direction quicker than if a symmetrical distribution occurred simultaneously. The collective behaviour of the group keeps the box moving even in the event one robot fails due to its quick replacement by a neighbouring robot.

5.6 Related Work

Within several different disciplines, there have been a number of research projects aimed at achieving some form of collective behaviour with a group of autonomous agents capable of sensing their immediate environment. This section briefly surveys some of the more relevant work in simulating collective behaviour.

5.6.1 Computer Graphics

Within the Computer Graphics literature, there is an animation technique referred to as Stimulus Response (S-R) animation. S-R animation is computer animation achieved by giving objects in a simulated environment rudimentary sensory capabilities and the ability to respond to other objects in the environment (Yang and Ware, 1989). The mapping between sensors and effectors provides for a behaviourally-controlled animation. This technique is being explored by a number of researchers (Reynolds, 1987; Wilhelms and Skinner, 1989; Yang and Ware, 1989).

The work of Craig Reynolds (1987) is particularly relevant to our approach. Reynolds has simulated the flight of a flock of birds using a distributed behavioural model which assumes a flock is simply the result of the interaction between the behaviours of individual birds. Each simulated bird is implemented as an independent actor that navigates according to its local perception of the dynamic environment. To build a simulated flock, Reynolds started with a bird model that supported geometric flight and added three behaviours that led to simulated flocking. Those behaviours were: Collision Avoidance, avoid collisions with nearby flockmates; Velocity Matching, attempt to match velocity with nearby flockmates; and Flock Centering, attempt to stay close to nearby flockmates.

Reynolds' approach differs from the one presented in this thesis in the method used to implement the flock centering behaviour. Flock centering causes the bird to fly in a direction that moves it closer to the centroid of the nearby birds. Since our physical robots cannot be equipped with a sensor that would allow for the calculation of the centroid we can not employ this technique in simulation. Our FOLLOW behaviour has a very local perception only and coupled with a behaviour-preference causes the herding behaviour similar to Reynolds' flocking.

5.6.2 Simulation of Adaptive Behaviour

An interesting approach to AI called the *Animat* approach involves simulating and understanding complete animal-like systems. The approach advocates gradually building up to human intelligence (Wilson, 1991). Within this literature are a few studying Collective Behaviour. Theraulaz et al. (1991) propose a simplified model for functional self-organization and how such a model could be applied in the coordination and self-organization of groups of interacting robots with simple local computational properties (for another example of collective behaviour see Deneubourg et al., 1991). Their model is also based on the study of social insects—wasp colonies in particular.

The model they present is based on their observations of task assignment in wasp colonies and describes the task assignment process in a hierarchically structured society in which two types of interaction control individual behaviour. The first is a *hierarchical type* of interaction in which the individuals of the society are ordered in a dominance structure with the highest ranking member designated the a individual. The second type of interaction is a *trophic type* of interaction controlling relationships between individuals and the environment. Trophic interaction represents the colony's demand for food and care, and varies an individual's response threshold to this type of stimulation. This eventually leads to a task specialization among individual colony members similar to a caste structure. Their model differs from ours in that we do not have a hierarchical structure for our populations; all individuals are considered the same. Our model also has no mechanism that varies the sensor response thresholds and therefore, no caste structure emerges from our system.

5.7 Summary

In this chapter we have presented our robot population simulator SimbotCity and its architecture for constructing and testing collective tasks. Simulation allows us to test the control strategies for populations of task achieving behaviour-based robots. Our approach is to employ a group of homogeneous robots controlled by a set of simple sensor driven reflexive behaviours whose actions are dependent on the current state of the environment in which the robot is situated. The resulting emergent global behaviour is a result of the individual robot's ability to sense its progress, via negative feedback, towards the collective goal.

In the next chapter, we present our physical implementation of the box-pushing task and describe the system's collective behaviour.

Chapter 6

Box-Pushing Robots: A Collective Task Implementation

6.1 Introduction—Implementing the Theory

In order to test the control strategies, developed in simulation, in the real world we have constructed a system of five identical behaviour-based mobile robots capable of achieving simple collective tasks without centralized coordination or the use of explicit communication (see figure 19). Control of the group of robots is accomplished by having each robot work toward a common goal. A simple form of cooperation among the robots is achieved by ensuring they do not interfere with each other. Each robot is autonomous and equipped with sensors for detecting both the goal and obstacles.

In this chapter we present a simple collective task implementation and discuss its relevance to the proposed control theory. We begin in section 6.2 by outlining our objectives in constructing robots and present the collective task we have demonstrated. In section 6.3 we present an overview of the robot's architecture. In section 6.4 we explain the architecture's implementation. Section 6.5 discusses the experimental results obtained in demonstrating the collective task. Finally, in section 6.6 the results are compared with the stated objectives and suggestions for other collective tasks are examined.

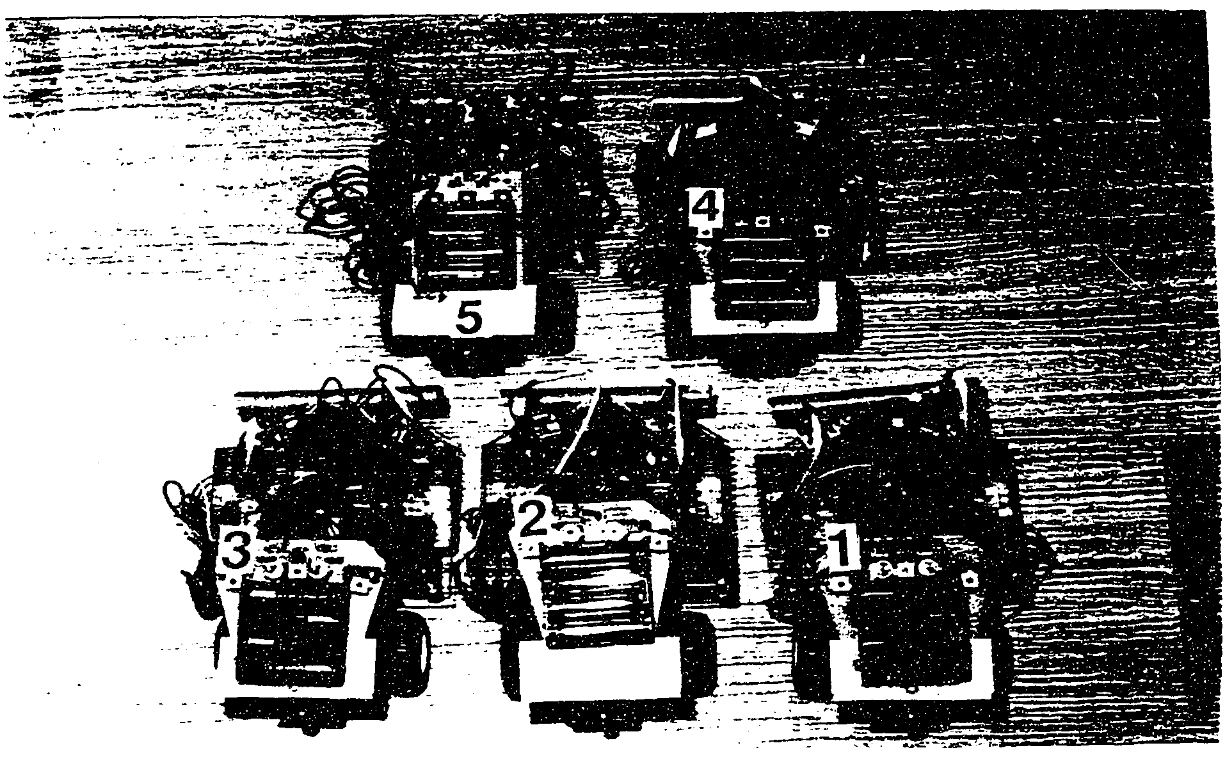

Figure 19: These five identical autonomous mobile robots were designed to locate and collectively push a box without centralized coordination or the use of explicit communication.

6.2 Objectives in Constructing Robots

Building physical robots and testing their performance and dynamic interaction in the real world is the ultimate test for any *proposed* system. In doing so, we hope to discover how well the proposed control model does or does not work. By having already tested the control strategies in simulation we have lessened the burden and increased the likelihood for success. However, as the inherent uncertainties of a dynamic real world rears its ugly head, we shall be prepared to meet them soldering iron in hand.

The collective task we chose to implement is box-pushing. The task is such that it cannot be accomplished by one robot because the box is too heavy to push; therefore, the collective effort of several robots is necessary. To accomplish this task the robots must locate the box; move toward it while avoiding collisions with other robots; distribute themselves along a side and push. In doing so, we hope to test the first lemma of our theory, namely, group behaviour may result by executing a common goal oriented collective task. We also hope to test the simple cooperative strategy of non-interference thereby verifying its usefulness in collective behaviour.

6.3 The Architecture of a Box-Pushing Robot

The robot's architecture consists of GOAL and AVOID behaviour modules. The GOAL behaviour functions to locate and guide the robot toward the box, while the AVOID behaviour handles obstacle avoidance including other robots. The control architecture is illustrated in figure 20.

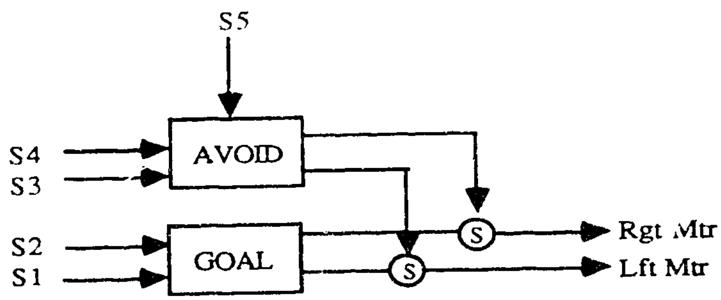

Figure 20: The box-pushing robot's control architecture used to control two wheel motors.

Each robot is equipped with five sensors. Sensors S1 and S2 are used by the GOAL behaviour to locate the box. Sensors S3 and S4 provide forward looking obstacle detection and sensor S5 is used to detect lack of progress toward the goal and is treated as an avoid response. A simple fixed priority between behaviours controls arbitration with the AVOID behaviour having the highest priority.

6.4 Implementing the Control Architecture

The control architecture is implemented using a combination of analog and digital circuits mounted on a plexiglass chassis. The architecture can be divided into three main functional units consisting of sensors, behaviours and actuators explained below.

Sensors S1 and S2 are photovoltaic cells capable of detecting the box which is equipped with a 100 watt light bulb. Sensors S3 and S4 are near infrared Light Emitting Diodes (LEDs) with a detection range from 1 to 18 inches. Sensor S5 is a switch which becomes active if the robot is pushed backwards. All sensors produce binary on or off signals.

Behaviours are implemented in combinational logic and are invoked when their associated sensor becomes active. The behaviours are reflexive and have no memory. Behaviour arbitration is handled with a simple combinational circuit assigning highest priority to the AVOID behaviour module. When the left photovoltaic sensor (S1) becomes active the GOAL behaviour activates the right wheel motor causing the robot to move toward the left. Likewise, when the right photovoltaic sensor (S2) becomes active the GOAL behaviour activates the left wheel motor causing the robot to move toward the right. When both sensors become active the GOAL behaviour activates both motors causing the robot to move straight forward. In this manner the robot can locate and move toward the box.

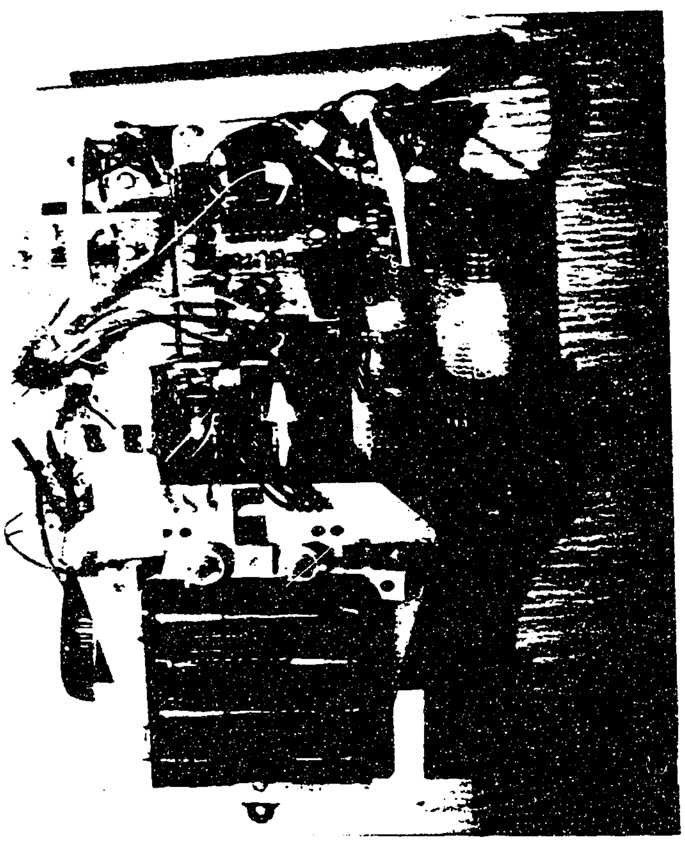

Figure 21: Close-up of an individual box-pushing robot.

The AVOID behaviour is invoked when any one of the two obstacle sensors S3 and S4 are active. Sensors S3 and S4 are located on the left and right side of the robot respectively (see figure 21) and sense obstacles within 10 inches of each side of the robot in a 40° cone. Activation of the left sensor causes the robot to move right; activation of the right sensor causes the robot to move left; activation of both sensors cause the robot to move right. Sensor S5 detects when the robot is being pushed backwards and therefore not making progress towards the goal and invokes the AVOID behaviour to turn the robot away from the box.

Each robot has two actuators, a left and right wheel motor used to provide mobility. Steering is accomplished by actuating one motor at a time. This allows the robot to turn left, right and forward. Backward motion is accomplished by reversing the polarity to the motors. An idle wheel on the front balances the tripod wheel configuration.

6.5 Demonstrating Group Behaviour

All demonstrations designed to test the system were video recorded. The first of these tests were performed on each individual robot in order to roughly calibrate the sensors to the same values. Tests involving the robots were conducted by starting the robots in an initial configuration, much the same way as in simulation. By adjusting the goal sensors to respond to the light on the box only, the system could be started and stopped by turning the box light on and off. The following sequence of pictures were taken in this manner.

The robots were first tested individually. Figures 22-24 show a robot avoiding an obstacle on its way to the box (shown at the top of the picture). The infrared sensors have a range from 1 to 18 inches depending on the physical alignment of the transmitter and receiver pair. Obstacle detection is accomplished by transmitting a directed modulated light signal in the near infrared spectrum and sensing its reflection with a phototransistor receiver. This method of obstacle detection is not completely accurate due to the reflectance property of some surfaces. The technique was sufficient for obstacle avoidance in the robots providing the velocity at which they travelled allowed sufficient time to negotiate a turn. One method of increasing the sensing range was to cover each robot in retro-reflecting tape.

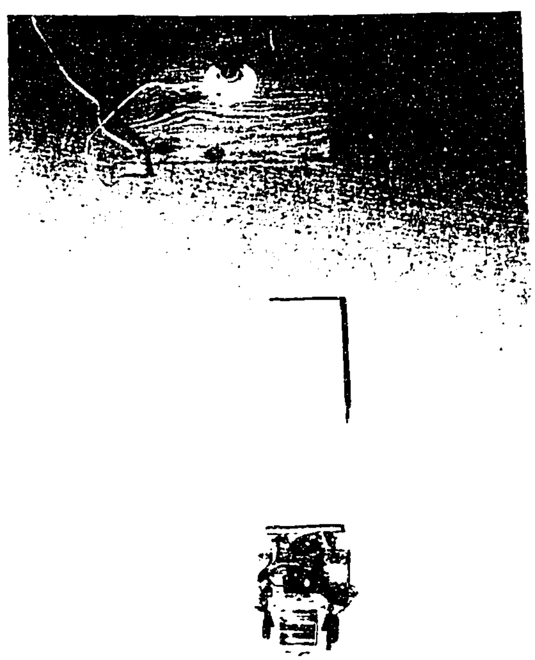

Figure 22: The first of a three picture sequence showing a single robot avoiding an obstacle on its way to the goal.

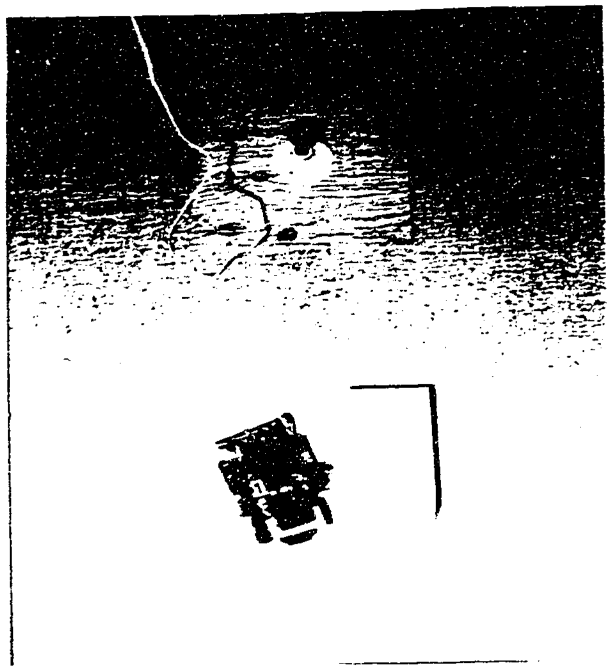

Figure 23: The second picture of three showing the robot avoiding an obstacle.

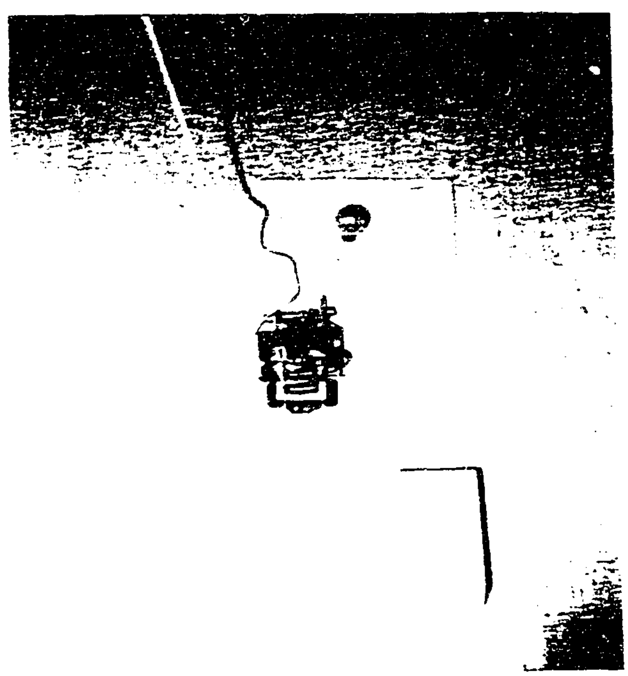

Figure 24: The last picture in a sequence of three showing obstacle avoidance.

Next the robots were tested in a group of three (figures 25-27). The initial configuration was set so the robots would collide if they continued on their original course. As they progressed toward the goal the AVOID behaviour kept the robots from colliding for the majority of the time. Collisions occurred whenever the sensors missed an oncoming robot. Reliability of the AVOID behaviour could be increased by adding additional infrared sensors.

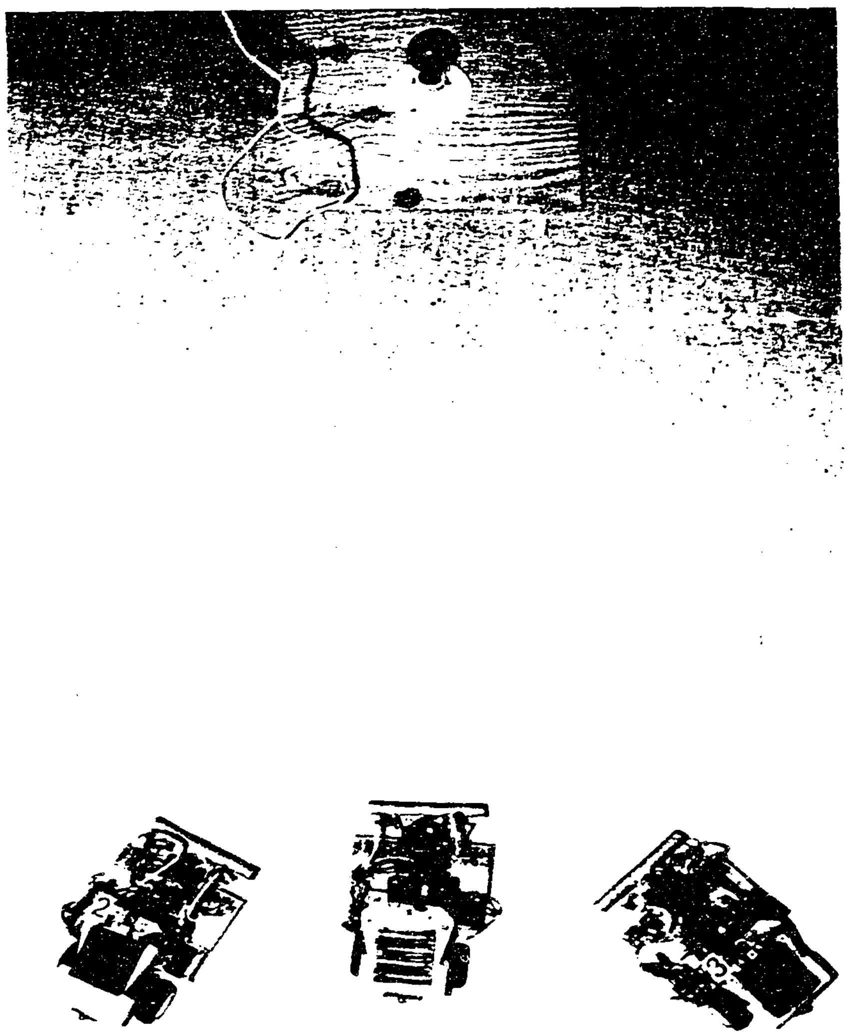

Figure 25: The first picture of three showing a group of three robots moving towards the goal.

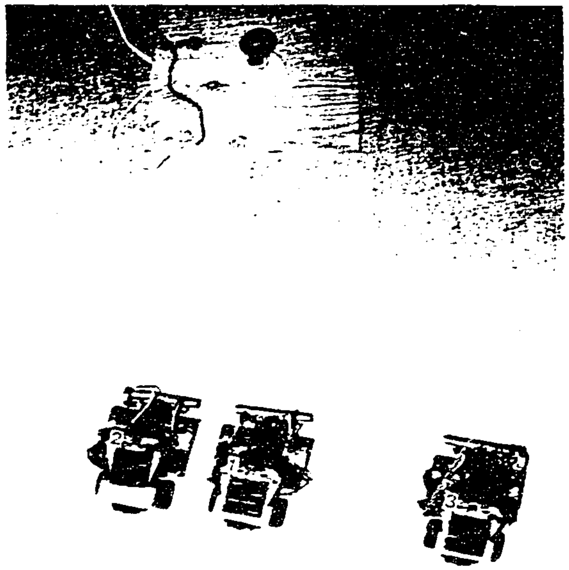

Figure 26: The second picture of three showing a group of three robots moving towards the box.

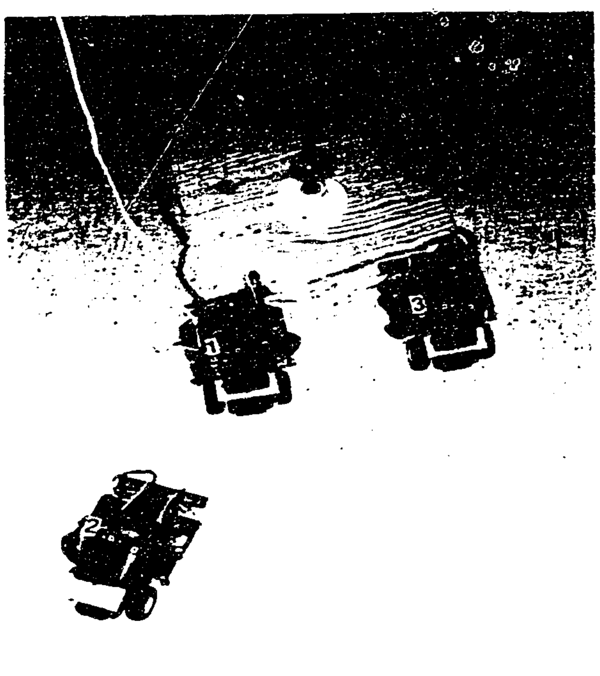

Figure 27: The last picture in a sequence of three showing the robots pushing the box.

The remaining demonstrations were conducted with all five robots in a variety of initial configurations. The robots converged on the goal and pushed the box in a number of directions depending on how many robots were on a given side.

6.6 Summary

In this chapter we have presented a simple collective task implementation which uses one of the five suggested methods for controlling a group of autonomous mobile robots. The approach we implemented involves controlling a group of five physical robots working toward a common goal. The collective task is accomplished without centralized coordination or the use of explicit communication among the robots. The described system demonstrates a simple form of cooperation takes place among robots that do not interfere with one another.

An important feature of the system is that simple reflexive behaviours can be used to control the individual robot in a goal directed manner using equally simple binary sensors. The behaviours and their arbitration mechanism are constructed using simple combinational logic. An important implication of this simplicity is that the control architecture could be scaled down to fit on a small silicon chip. This would allow for the creation of a large number of cost effective robots to be used in areas too tiny for more traditional robots. Finally, the system demonstrates the feasibility of controlling populations of autonomous robots without the need for explicit communication.

Chapter 7
Conclusion

7.1 Summary

Research in behaviour-based robotics has led to radically different architectures for controlling autonomous robots. These new architectures emphasize a more direct coupling of perception to action and a dynamic interaction with the environment resulting in systems with an emergent functionality. Systems that chose to employ this methodology must be designed in a way that make use of an interaction loop between the system and the environment which ultimately converges towards the desired goal. Most research projects have concentrated on designing single autonomous robots capable of achieving a simple insect-like intelligence. Useful tasks may be accomplished with these simple behaviour-based control mechanisms provided multiple robots are organized into collections of task achieving populations.

The research described in this thesis attempts as the first step to propose a control theory suitable for controlling populations of behaviour-based robots. Our approach to controlling multiple robots involves the use of group behaviours which may be invoked using several sensory-based mechanisms. The strategies proposed have resulted from the study of social insects which exhibit collective task achieving behaviours. To test our control theory we created a simulator, called *SimbotCity*, which allowed us to create configurations of multiple robots designed to achieve a collective task. Once satisfied the control strategy was feasible, we then constructed a system of five physical robots designed to accomplish a simple collective task without any centralized coordination or the use of explicit communication. The approach, employed to control the group of five robots, involved having the robots work toward a common goal. Using non-interference as a simple form of cooperation the robots were able to collectively locate and push a box in their environment. The system demonstrates the feasibility of the proposed control theory warranting its further investigation.

7.2 Future Research

There are a number of areas of the described research that could be expanded upon beginning with the simulator, *SimbotCity*. In its current version (1.08) robots are constructed by coding the selected sensor, actuator and behaviours and then recompiling the system. A more general implementation allowing the user to interactively construct a robot by selecting its components would be more useful. Also, task definition should allow for a wider range of collective tasks to be explored by the user.

Group behaviour is the unique method the theory advocates for controlling populations of autonomous mobile robots. The system of physical robots explored the first method proposed for invoking group behaviour. This method involved using a common goal as a means of controlling the group of autonomous robots. The remaining four methods of group behaviour need to be tested in the real world.

The second method of invoking group behaviour involves keeping the robots together in herds. This may be accomplished by designing a suitable FOLLOW behaviour. The behaviour would require sensors that detect the presence of other robots, this may be accomplished using infrared transmitting beacons on each robot. Receivers on the robot would detect and guide the robot toward the herd. The group would then move as one mass capable of distributing itself over an area. This could be useful if the system of robots were being used to gather spatial data. The system could then be viewed as a flexible array of sensors capable of reconfiguring itself. Such a system would possess sensing capabilities similar to Moravec's Robot Bush (Moravec, 1988), a fictitious robot constructed in a tree-like fashion with hundreds of tactile sensors at its leaves.

The third method of invoking group behaviour involves using environmental cues. Collective tasks can be designed to allow the environment, in which the robots work, to provide the cue that invokes the group behaviour. This method requires careful study of the environmental changes as the task proceeds and focuses on the perceptual changes rather than the individual steps in the collective task. Task progression is then viewed as a sequence of behavioural steps, with each step effecting a perceivable change in the environment. These changes form the environmental cues necessary to invoke the next behavioural step in the task. Since the robots all perceive these changes simultaneously the method forms a basis for controlling the group.

The fourth method of invoking group behaviour entails having the robots become aware they are in a group. Special task achieving group behaviours then become active. The method could employ a sensor ring which surrounds the robot and is used to detect the presence of neighbouring robots. Robots in the middle of such a group would then invoke the group behaviour. This form of control would divide the robots into different classes in which robots within the group are assigned different functions.

The fifth method of invoking group behaviour uses autostimulation. This is a form of explicit communication in which the robot broadcasts a signal that invokes the group behaviour. This method would be useful in tasks requiring many possible decision branches for their execution. For example, using multiple robots to perform a search for an object, which once found could terminate the search. Other forms of explicit communication for group control could also be explored.

The research described in this thesis is intended to be an initial exploration into achieving tasks collectively using a system of multiple robots. As such, its primary goal was to examine the feasibility of the approach we have outlined in this report. The work has proven to be formidable, but the experiences of building robots which operate in the real world has been an enlightening one. Collective Robotics may one day prove to be a suitable tool in achieving useful tasks.

Bibliography

Agre, P. and Chapman, D. (1987). Pengi: An Implementation of a Theory of Activity. In *Proceedings of AAAI-87* pp. 268-272.

Agre, P. and Chapman, D. (1990). What Are Plans For? In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 17-34.

Anderson, T. and Donath, M. (1988a). Synthesis of Reflexive Behaviours for a Mobile Robot Based Upon a Stimulus-Response Paradigm, in *Proceedings of the 1988 SPIE Conference on Mobile Robots*, pp. 370-382.

Anderson, T. and Donath, M. (1988b). A Computational Structure for Enforcing Reactive Behaviour in a Mobile Robot, in *Proceedings of the 1988 SPIE Conference on Mobile Robots*, pp. 198-211.

Anderson, T. and Donath, M. (1990). Animal Behavior as a Paradigm for Developing Robot Autonomy. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 145-169.

Arkin, R. (1987). Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior, *IEEE Conference Robotics and Automation*, Mar. pp. 264-271.

Arkin, R. (1990). Integrating Behavioral, Perceptual, and World Knowledge In Reactive Navigation. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 105-122.

Beer, R. (1990). *Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology*. Academic Press, Inc.

Beer, R., Chiel, H. and Sterling, L. (1990a). A Biological Perspective on Autonomous Agent Design. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 169-186.

Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press.

Brooks, R. (1986). A Robust Layered Control System for a Mobile Robot, in *IEEE Journal of Robotics and Automation, RA-2 , pp*. 14-23.

Brooks, R. and Flynn, A. (1989). Fast, Cheap and Out of Control. MIT AI Memo 1182.

Brooks, R. (1990). Elephants Don't Play Chess. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 3-15.

Brooks, R. (1991). Intelligence Without Reason. A.I. Memo No. 1293, MIT AI Lab.

Brooks, R. (1991a). Challenges for Complete Creature Architectures. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 434-443.

Camhi, J.M. (1984). *Neuroethology*. Sinauer Associates, Sunderland, MA.

Chandrasekaran, B. (1990). Editor's note in *Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology*, Perspectives in AI Vol. 6, Academic Press.

Chavin, R. (1964). Expériences sur L'Apprentissage par Équipe du Labyrinthe chez *Formica polyctena*. In *Insectes Sociaux*, 11(1): pp. 1-20.

Coderre, B. (1988). Modelling Behavior in Petworld. In C. Langton (Ed.), *Artificial Life*. Addison-Wesley Publishing Company.

Connell, J. (1990). *Minimalist Mobile Robotics: A Colony-style Architecture for an Artificial Creature*, Perspectives in AI Vol. 5, Academic Press.

Dario, P., Ribechini, F., Genovese, V. and Sandini, G. (1991). Instinctive Behaviors and Personalities in Societies of Cellular Robots. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*. pp. 1927-1932.

Deneubourg, J-L. Goss, S., Pasteels, J.M., Fresneau, D. and Lachaud, J-P. (1987). Self-Organization Mechanisms in Ant Societies II: Learning in Foraging and Division of Labor. In *From Individual to Collective Behavior in Social Insects*, Pasteels, J.M. and Deneubourg, J-L. (eds.), (*Experientia* Supplement, *vol 54*). pp. 177-196.

Deneubourg, J-L. and Goss, S. (1989). Collective Patterns and Decision-Making. In *Ethology, Ecology and Evolution*. 1: pp. 295-311.

Deneubourg, J-L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C. and Chrétien, L. (1991). The Dynamics of Collective Sorting Robot-Like Ants and Ant-Like Robots. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 356-363.

Frisch, K. von (1949). Die Polarisation des Himmelslichtes als orientierender Faktor bei den Tänzen der Bienen. In *Experientia*, 5(4): pp. 142-148.

Frisch, K. von (1967). *The Dance Language and Orientation of Bees* (tr. Chadwick). Belknap Press of Harvard University Press.

Giralt, Chatila and Vaisset (1984). An Integrated Navigation and Motion Control System for Multisensory Robots, in *Robotics Research 1*, Brady and Paul (eds), MIT Press, pp. 191-214.

Grassé, Pierre-Paul (1959). La Reconstruction du Nid et les Coordinations Interindividuelles chez *bellicositermes natalensis* et *Cubitermes* sp. La Théorie de la Stigmergie: Essai d'interprétation du comportement des Termites Constructeurs. In *Insectes Sociaux*, 6(1): pp. 41-83.

Grassé, Pierre-Paul (1967). Nouvelles Expériences sur le Termite de Müller (*Macrotermes mülleri*) et Considérations sur la Théorie de la Stigmergie. In *Insectes Sociaux*, 41(1): pp. 73-102.

Goebel, R. (1988). *Representation and Reasoning: A Logical Introduction to Artificial Intelligence,* Randy Goebel and David Poole, 1988.

Hangartner, W. (1969). Carbon Dioxide, A Releaser For Digging Behavior in Solenopis Geminata (Hymenop [illegible] ormicidae). In *Psyche,* 76(1): pp. 58-67.

Haskins, C.P. and Enzmann, E. [illegible] 38). Studies of Certain Sociological and Physiological Features in the Formicidae. In *Annals of the New York Academy of Sciences,* 37(2): pp. 97-162.

Hartley, R. (1991). Experiments With The Subsumption Architecture. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation,* April, pp. 1652-1658.

Heran, H. (1952). Untersuchungen über den Temperatursinn der Honigbiene (*Apis mellifica*) unter besonderer Berücksichtigung der Wahrnehmung strahlender Wärme. In *Zeitschrift für Vergleichende Physiologie,* 34(2): pp. 179-206.

Hinde, R.A. (1966). *Animal Behaviour: A Synthesis of Ethology and Comparative Psychology.* McGraw-Hill Book Co.

Hölldobler, B., Möglich, M. and Maschwitz, U. (1974). Communication By Tandem Running in the Ant Camponotus Sericeus. In *Journal of Comparative Physiology,* 90(2): pp. 105-127.

Hopfield, J. (1984). Neurons With Graded Responses Have Collective Computational Properties Like Those of Two-State Neurons. In *Proc. Natl. Acad. Sci.* 81. pp. 3088-3092.

Jander, R. (1957). Die optische Richtungsorientierung der Roten Waldameise (*Formica rufa* L.). In *Zeitschrift für Vergleichende Physiologie,* 40(2): pp. 162-238.

Kaelbling, L. (1987). An Architecture for Intelligent Reactive Systems. In *Proceedings of the 1986 Workshop on Reasoning about Actions and Plans,* pp.395-410.

Kaelbling, L. and Rosenschein, S. (1990). Action Planning In Embedded Agents. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back,* Maes, P. (ed.), MIT Press. pp. 35-48.

Kalmus, H. (1937). Vorversuche über die Orientierung der Biene im Stock. In *Zeitschrift für Vergleichende Physiologie,* 24(2): pp. 166-187.

King, G.E. (1933). The Large Glands in the Worker Honey Bee. A Correlation of Activity With Age and With Physiological Functioning. Abstract of Ph.D. Thesis, University of Illinois, pp. 20.

Langton, C. (1988). Artificial Life. In *Proceedings of the First Conference on Artificial Life*, pp. 1-47.

Lindauer, M. (1952). Ein Beitrag Zur Frage der Arbeitsteilung im Bienenstaat. In *Zeitschrift für Vergleichende Physiologie*, 34(4): pp. 299-345.

Lindauer, M. (1954). Dauertänze im Bienenstock und ihre Beziehung zur Sonnenbahn. In *Naturwissenschaften*, 41(21): pp. 506-507.

Lindauer, M. and Martin, H. (1963). Über die Orientierung der Biene im Duftfeld. In *Naturwissenschaften*, 50(15): pp. 509-514.

Lorenz, K. Z. (1981). *The Foundations of Ethology*. New York: Simon and Schuster.

Maes, P. (1990). *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*. Pattie Maes (Ed.), MIT Press.

Markl, H. (1962). Borstenfelder an den Gelenken als Schweresinnesorgane bei Ameisen und anderen Hymenopteren. In *Zeitschrift für Vergleichende Physiologie*, 45(5): pp. 475-569.

Martin, H. (1964). Zur Nahorientierung der iene im Duftfeld. Zugleich ein Nachweis für die Osmotropotaxis bei Insekten. In *Zeitschrift für Vergleichende Physiologie*, 48(5): pp. 481-533.

Menzel, R. (1968). Das Gedächtnis der Honigbiene für Spektralfarben. I. Kurzzeitiges und langzeitiges Behalten. In *Zeitschrift für Vergleichende Physiologie*, 60(1): pp. 82-102.

Meyer, J-A. and Guillot, A (1991). Simulation of Adaptive Behavior in Animats: Review and Prospect. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 2-14.

Minsky, M. (1985). *The Society of Mind*. New York: Simon and Schuster.

Moravec, H. (1981). *Robot Rover Visual Navigation*. UMI Research Press.

Moravec, H. (1988). *Mind Children*. Harvard University Press.

Moser, J.C. (1970). Pheromones of Social Insects. In *Control of Insect Behavior by Natural Products*, Wood, D., Silverstein, R. and Nakajima, M. (Eds.), Academic Press, pp. 161-178.

Moyson, F. and Manderick, B. (1988). The Collective Behavior of Ants: An Example of Self-Organization in Massive Parallelism. *Technical Report* Vrije Universiteit Brussels AI Memo 88-7.

Nilsson, N, ed. (1984). *Shakey the Robot*. A.I. Center Technical Note 323. Menlo Park: SRI International.

Oster, G.F. and Wilson, E.O. (1978). *Caste and Ecology in the Social Insects*. Princeton University Press.

Pasteels, J.M., Deneubourg, J. and Goss, S. (1987). Self-Organization Mechanisms in Ant Societies: Trail Recruitment To Newly Discovered Food Sources. In *From Individual To Collective Behavior in Social Insects*, Pasteels, J.M. and Deneubourg, J-L. (eds.), (*Experientia* Supplement, *vol 54*). pp. 155-175.

Payton, D.W. (1986). An Architecture for Reflexive Autonomous Vehicle Control. In *Proc. IEEE Conf. on Robotics and Automation*, pp. 1838-1845.

Payton, D.W., Rosenblatt, J.K., and Keirsey, D. (1990). Plan Guided Reaction. In *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 6, Nov./Dec. pp. 1370-1382.

Payton, D.W. (1990a). Internalized Plans: A Representation For Action Resources. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, Maes, P. (ed.), MIT Press. pp. 89-103.

Peacock, A.D., Hall, D.W., Smith, I.C.and Goodfellow, A. (1950). The Biology and Control of The Ant Pest—Monomorium Pharaonis, *Miscellaneous Publications*, Department of Agriculture for Scotland, 17: pp. 1-51.

Peacock, A.D., Waterhouse, F.L.and Baxter, A.T. (1955). Studies in Pharaoh's Ant—Monomorium Pharaonis. In *Entomologist's Monthly Magazine*, 91: pp. 37-42.

Reynolds, C.W. (1987). Flocks, Herds, and Schools: A Distributed Behavioural Model. In *Computer Graphics*, Vol. 21, No. 4. pp. 25-34.

Santschi, F. (1911). Observations et Remarques Critiques sur le Mécanisme de L'Orientation chez les Fourmis. In *Revue Suisse de Zoologie,* 19(13): pp. 303-338.

Schmidt, Anneliese, (1938). Geschmacksphysiologische Untersuchungen an Ameisen. In *Zeitschrift für Vergleichende Physiologie,* 25(3): pp. 351-378.

Schneirla, T.C. (1940). Further studies on the Army-Ant Behavior Pattern: Mass-Organization in the Swarm-Raiders. In *Journal of Comparative Psychology,* 29(3): pp. 401-460.

Schneirla, T.C. (1946). Ant Learning as a Problem in Comparative Psychology. In *Twentieth Century Psychology* Harriman, P. (ed.), New York. pp. 276-305.

Schneirla, T.C. (1956). The Army Ants. In *Reportt of The Smithsonian Institution for 1955,* pp. 379-406.

Seeley, T.D. and Levien, R.A. (1987). Social Foraging by Honeybees: How a Colony Tracks Rich Sources of Nectar. In *Proceedings 10th International Congress.* IUSSI, Müchen.

Sekiguchi, M., Nagata, S. and Asakawa, K. (1989). Behavior Control For A Mobile Robot By Multi-Hierarchical Neural Network. In *Proc. IEEE Conf. on Robotics and Automation,* pp. 1578-1583.

Steels, L. (1987). Self-Organization Through Selection. *Technical Report* Vrije Universiteit Brussels AI Memo 87-5.

Steels, L. (1989). Cooperation Between Distributed Agents Through Self-Organization. *Technical Report* Vrije Universiteit Brussels AI Memo 89-5.

Steels, L. (1990). Cooperation Between Distributed Agents Using Self-Organization. In Y. Demazeau and J-P. Muller (1990) *Decentralized AI.* North-Holland Pub.

Steels, L. (1991). Towards a Theory of Emergent Functionality. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior,* Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 451-461.

Simon, H.A. (1970). *The Sciences of the Artificial,* MIT Press.

Stuart, A.M. (1967). Alarm, Defence, and Construction Behavior Relationships in Termites (*Isoptera*). In *Science,* 156: pp. 1123-1125.

Stuart, A.M. (1969). Social Behavior and Communication. In *Biology of Termites*. Krishna, K. and Weesner, F.M. (eds.), Vol. I. pp. 193-232.

Theraulaz, G., Goss, S., Gervet, J. and Deneubourg, J-L. (1991). Task Differentiation in Polistes Wasp Colonies: A Model for Self-Organizing Groups of Robots. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 346-355.

Travers, M. (1988). Animal Construction Kits. In C. Langton (Ed.), *Artificial Life*. Addison-Wesley Publishing Company.

Voss, Christianne, (1967). Über das Formensehen der roten Waldameise (*Formica rufa-Gruppe*). In *Zeitschrift für Vergleichende Physiologie*, 55(3): pp. 225-254.

Vowles, D.M. (1950). Sensitivity of Ants to Polarized Light. In *Nature*, 165: pp. 282-283.

Vowles, D.M. (1954). The orientation of Ants. I. The Substitution of Stimuli. In *Journal of Experimental Biology*, 31(3): pp. 341-355.

Webster (1988). *The New Lexicon Webster's Encyclopedic Dictionary of the English Language*. Lexicon Publications, Inc.

Weiss, K. (1953). Versuche mit Bienen und Wespen in farbigen Labyrinthen. In *Zeitschrift für Tierpsychologie*, 10(1): pp. 29-44.

Weiss, K. (1957). Zur Gedächtnisleistung von Wespen. In *Zeitschrift für Vergleichende Physiologie*, 39(6): pp. 660-669.

Wilhelms, J. and Skinner, R. (1989). An Interactive Approach to Behavioral Control. In *Proceedings of Graphics Interface 1989*, pp. 1-8.

Wilson, E.O. (1962). Chemical Communication Among Workers of the Fire Ant *Solenopsis saevissima*. In *Animal Behaviour*, 10(1-2): pp. 134-164.

Wilson, E.O. (1971). *The Insect Societies*, The Belknap Press of Harvard University Press.

Wilson, E.O. and Hölldobler, B. (1990). *The Ants*, The Belknap Press of Harvard University Press.

Wilson, S.W. (1991). The Animat Path to AI. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior,* Meyer,J-A. and Wilson, S.W. (eds.), MIT Press. pp. 15-21.

Yang, S. H. and Ware, C. (1989). ESCIM: A System for the Investigation of Meaningful Motion. In *Proceedings of Graphics Interface 1989,* pp. 9-13.

Yuta, S. and Premvuti, S. (1991). Consideration on Cooperation of Multiple Autonomous Mobile Robots, *IEEE/RSJ International Workshop on Intelligent Robots and Systems.* pp. 545-549.