5-2010

Design of Adaptive Collective Foraging in Swarm Robotic Systems

Hanyi Dai
Western Michigan University

DESIGN OF ADAPTIVE COLLECTIVE FORAGING IN SWARM ROBOTIC SYSTEMS

by

Hanyi Dai

A Dissertation
Submitted to the
Faculty of The Graduate College
in partial fulfillment of the
requirements for the
Degree of Doctor of Philosophy
Department of Electrical and Computer Engineering
Advisor: Frank Severance, Ph.D.

Western Michigan University
Kalamazoo, Michigan
May 2010

DESIGN OF ADAPTIVE COLLECTIVE FORAGING IN SWARM ROBOTIC SYSTEMS

Hanyi Dai, Ph.D.

Western Michigan University, 2010

Inspired by the collective behavior observed in biological colonies, swarm robotics is a new approach to design a distributed control algorithm in order to coordinate a group of simple robots performing a complex group task. With only limited computation and communication ability of individual robots, one of the challenges in designing such multi-robotic systems is to understand the effect of individual robots behavior on the group performance. This thesis dedicates the research to designing a set of local interaction and adaptation rule for individual robots so that optimized collective foraging performance can be achieved at group level.

The research starts with designing a computer simulation program to simulate collective foraging of multiple robots. A behavior based controller was used to design a robot that performs basic foraging behavior.

Inspired by the widely observed division of labor phenomenon in biological systems, the desired group behavior in collective foraging is to have an optimal division of labor between active foraging and resting among robots. Two variables, foraging threshold and foraging stimulus, were used to calculate the foraging probability for each robot. A set of local adaptation rules are designed to adapt these two variables in a

self-organized manner through the multiple interactions between robot and food source in the environment, as well as interactions between foraging robots. The results of the experiments show that the robot group with adaptation mechanisms not only achieve optimal group foraging performance, but also show robustness and flexibility of robot group to environment changes.

The study then extended to that a group of heterogeneous robots foraging in a more complex environment with two types of food. Desired robot group behavior requires an optimal task allocation of foraging robots on two types of food, and division of labor between foraging and resting. The interaction rules were upgraded for individual robots. The results of the experiments show that both division of labor and task allocation emerged at a group level. The robot group also demonstrates the ability to collectively perceive the changes in the environment and to guide the group toward foraging performance optimization.

ACKNOWLEDGMENTS

Many People have helped me during my study. First and for most, I would like to thank my advisor Prof. Frank Severance for his guidance and advise to complete this work. I really appreciate the support that Prof. Severance gave me in choosing the research direction and method. Along the way I have learned a lot from the discussion with him, both from formal meeting and informal chatting every time. Without the help from him, this thesis couldn't reach its final stage.

Special thanks to my committee: Prof. Sievers and Prof. Dong for their support and help. I appreciate for they spending time to review my work and give me good advice. I also would like to thank all faculties in ECE department for continue support during my studying and teaching at Western Michigan University, especially thank Dave Florida for continues help and encouragement during my study.

Finally, I would like to thank my parents, my brother, and my friends for their endless love and support during the whole period of my education. A special thank goes to my good friend Don Toppen, for providing continued support and encouragement during my stay in Unite State.

Hanyi Dai

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

List of Figures-Continued

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Introduction

The term "robot "was first used by Czech writer Karelapek in his play R.U.R. (Rossum's Universal Robots), published in 1920. In the play, a factory makes artificial people called robots who can think for themselves. Since the play, robots have become more advanced and sophisticated, and the word "robotics"was used to describe this field of study. Robotic systems, an automatic manufacturing process which reducing manual labor attached with it, are used extensively in almost every field these days.

Based on the various applications of robots today, robotic systems are classified mainly into three types. A manipulation robotic system, mobile robotic system and data acquisition robotic system. Manipulation robotic system is the most extensively used system in manufacturing industries. A mobile robotic system is usually an automated process that robots move from one place to another. The motion of the robotic system can be controlled autonomously and might have a pre-programmed destination from where the system might load or unload automatically. A Data acquisition robotic system is used for acquiring, processing, and transmitting important data used for generating various signals. Figure 1.1 shows the general classification of the robotic system. Earlier mobile robotic system mainly focused on the development of a Single

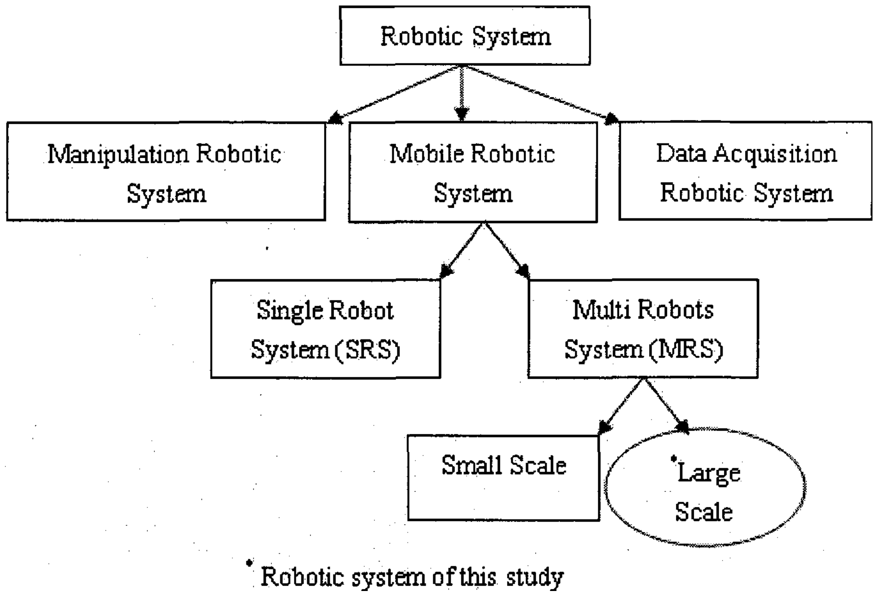


Figure 1.1 Classification of the Robotic System

Robot System (SRS). With the development of the computer technique and the application of a simpler behavior based control structure for the robot, Multi Robots Systems (MRS) have received increased attention in recent years.

MRS provides some potential advantages over a Single Robot System (SRS). Certain tasks may be too complex to be accomplished by a single robot no matter how capable the single robot is. For example, in some situation, task needs to be accomplished in a large space at the same time or the task needs simple cooperation by more than one robot. SRS can not accomplish such task on its own, which makes the SRS often only feasible to the smaller-scale, simpler applications. Sometimes although a single robot can finish the required task if given enough time, it is better to use multiple robots to complete the task in order to improve the performance of the system. In addition, designing and using simple robots in a MRS may be potentially easier, cheaper, and more flexible and robust to the changes by taking advantage of inherent parallelism and redundancy than using one single powerful robot.

However, having multiple robots in a system makes a design procedure more challenging. Clearly, MRS inherits all the challenges in designing a SRS so that each robot can perform a desired task. More issues need to be addressed in order to coordinate multiple robots to collectively complete a desired task. For exmple, what are the control architectures of the designed MRS (centralized or decentralized control)? Do the robots in the system behave identically (homogeneous) or differently (heterogeneous)? How can the robots resolve the resource conflict problem in a shared environment? What type of communication method (explicitly or implicitly) is necessary for robots to communicate in order to perform the desired tasks? Without all robots coordinated working toward a same goal, a MRS can be even less effective than a carefully designed SRS.

As a relatively new approach to design a larger scale MRS, swarm robotics provide a decentralized control structure to coordinate a large number of robots working towards a desired group task. Swarm Robotics is a study inspired by the highly coordinated and adaptive collective group behavior observed in biological systems of social insects. Figure 1.2 shows the relation between other MRS and swarm robotic system. Swarm robotic system is often composed of a large number of simple robots which only have limited sense and communication ability. The coordination among the robots is achieved in a self-organized manner; the collective behavior of the group, which also

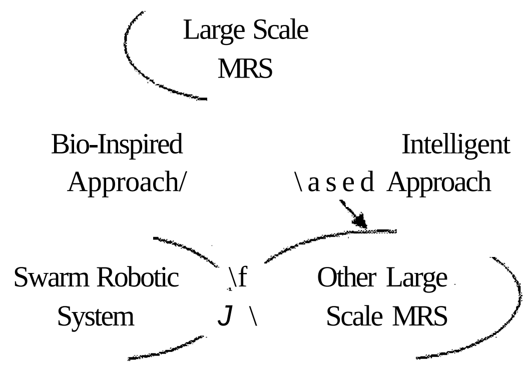


Figure 1.2 Swarm Intelligence Inspired MRS

is the desired group task, comes from the local interactions among robots, and between the robots and the environment. Dorigo and Sahin [1] highlight the main features of a swarm robotic system beyond other multi-robotic systems:

(i) It consists of a large number of robots;

(ii) Individual robots in the system are relatively incapable or inefficient to complete the task on their own;

(iii) Robots in a swarm robotic system only have local and limited sensing and communication abilities.

Despite the limited sensing, communication and computation abilities of individual robots in a swarm robotic system, the most exciting property of such system is that coordinated complex behaviors are shown at the group level. Inspired by its biological counterpart, a swarm robotic system potentially offers following benefits:

- **Flexibility** - Robot group shows the ability to respond to dynamic challenges in the environment. Because implicit communication is often used in a swarm robotic system, individuals in the system normally respond to the environmental changes resulting from external perturbations, just as if it is a modification of the environment caused by the activity of the other robots.

- **Robustness -** Robot group shows robustness to the loss or malfunction of individual robots. By distributing a desired task across a group of simple robots, the robot group can exhibit a high fault tolerance. Because of a large number of individuals in the group, the loss or malfunction of an individual can be compensated by another one, as the behavior of each robot can be duplicated by its peers. Besides, the simplicity of the individual robot suggests a smaller possibility of failure when performing the task.

- **Scalability -** The robot group can function with a size from a few units to thousands of units. As each robot in a swarm robotic system has only local sensing and communication ability, a robot only interacts locally with other robot or its environment. As a result, each robot uses exactly the same control architecture.

- **Emergence -** A complex coordinate group behavior emerges from the local interactions between the robots and between the robots and the environment. Importantly, the group behavior is not explicitly coded within the rules that each individual follows. Rather, it emerges as a result of the interactions of the individual behaviors in a self-organized manner.

In a swarm robotic system, the desired complex behavior which are designed to show at the group level cannot be observed at the individual level. Small changes in local behavior of individuals could lead to a total difference in group-level behavior. Despite the exciting benefits a swarm robotic system provides over other multi-robotic systems, one of the challenges in designing such a swarm robotic system is to understand how individual robot behavior affects the group behavior. This thesis was dedicated to the design of local interaction and adaptation rule for individual robots so that an optimized collective foraging behavior can emerge at the group level.

Collective foraging is a classic group task in the study of multi-robotic system. Foraging behavior of a robot includes searching in an area for food and retrieving food (energy) back to a designed place. In fact, the foraging task can be accomplished by a single robot if given enough time, but a group of robots working together could potentially improve the foraging performance. It is especially hard for one robot foraging in a large area, considering energy efficiency, so multiple robots can be considered in a foraging task. Foraging robots bring energy back to the group only if they successfully retrieve food back with positive net energy. More foraging robots do not necessarily increase the net energy of the group considering energy cost by more robots moving around. Considering the negative impact of interference between foraging robots due to robots competing for space (overcrowding), net energy of the group from foraging activity will not increase monotonically with the number of foraging robots. From the collective foraging efficiency point of view, the number of active foraging robots in a given environment need to be optimized in order to improve foraging performance. The desired group behavior is to have an optimal number of active foraging robots.

Motivated by the group goal of having an optimal number of active foraging robots and inspired by a widely observed division of labor in social insect colonies, the desired group behavior for the collective foraging is to have an optimal division between active foraging robots and resting robots. In this study, we are particularly interested in designing a set of local interaction and adaptation rules for individual robot so that optimal division of labor can emerge at robot group level.

Individual robots can switches task between foraging and resting according to its foraging probability. A robot with a high foraging probability forage more often. A set of adaptation rules are designed for individual robots to adapt the foraging probability based solely on the results from multiple interactions between foraging robots and between robot and environment. Division of labor between resting and foraging at group level emerge in a self-organized way. As a result, foraging performance of the design group is highly improved. Designed group also shows the flexibility to the environment changes, the robustness to individual foraging robots failure as well as the ability to function at different group sizes.

Three metrics have been designed to measure foraging performance of the design group in this study: energy efficiency, time efficiency, and retrieval efficiency, all at the group level. Foraging robots retrieve food energy back but also consume energy while they forage in the field. Net energy is defined as the difference between food energy income and energy cost by the group. Energy efficiency which measures the percentage net energy can be retrieved home from available food energy source in environment through collective foraging activity. Time efficiency measures how fast a robot group to retrieve net energy. Retrieval efficiency measures the percentage of food energy in an environment which can be retrieved by the robot group.

Although experiments using real robots provide realistic environments and results for robotic study, it is more common to use computer simulations in the study of swarm robotics. A swarm robotic system normally includes a large number of robots to exhibit the collective behavior; it may take a long time to build and debug each piece of hardware required for the experiments. Simulation often is much easier, less expensive, and runs much faster than real robots. As matter of fact, all research of swarm robotics use computer simulation as a necessary validation tool at earlier design stage.

We develop a new multi-robots foraging simulation program in order to validate the proposed design. The simulation program has class modules for robots, sensors, foods and environments which can be set up according to simulation requirements. There are up to 12 robots available in the simulation program to perform collective foraging tasks. The simulation program has data acquisition functions which can record all the data related to group foraging performance into a separated tile for later data analysis. Figure 1.3 is a screen shot for a simulation of a group of robots collectively forage in a bounded environment. In first set of experiments, several robot groups with different populations collective forage in a given environment with a fixed food growth rate.

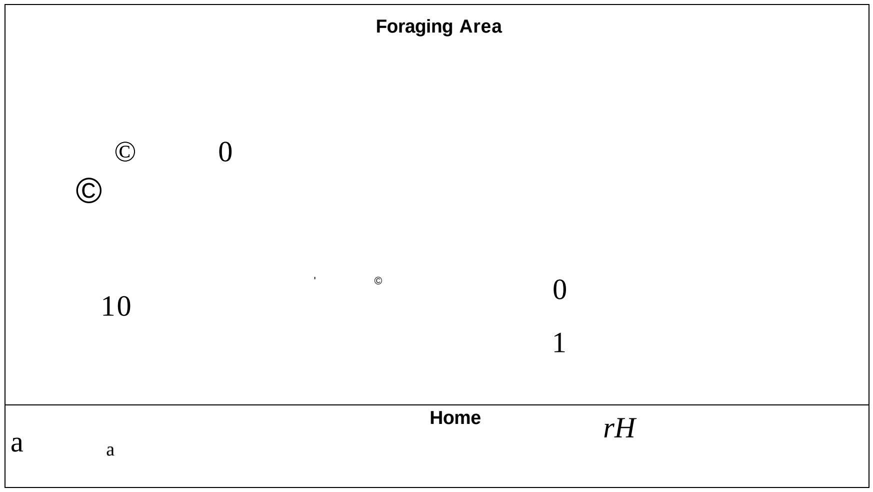


Figure 1.3 Robots Collective Foraging Simulations

Each robot follows the designed interaction and adaptation rules. Divisions of foraging robots and resting robots in design groups are confirmed at steady state in each foraging process first. Design group also shows highly improved energy and time efficiency.

A design group with 12 robots then collective forages in different environments with various food growth rates. Even food source information is not available to individual robot, through the multiple interactions between foraging robot and environment and between foraging robots, design group shows flexibility by have different number of foraging robots according to the foraging environment.

Design group also was tested in a dynamic foraging environment which changing food growth rates. Design group shows flexibility by adapting to the changes. Design group also shows robustness by recovering from dysfunction of some foraging robots.

Once the proposed design was validated in an environment with one type of food, we extended the interaction rules for a group of robots forage in a food source with more than one type of food. Individual robot can only retrieve one type of food at a time. Foraging robots behave differently according to the type of food it set to forage at the time. It became a heterogeneous group. The goal for the robot group is to retrieve as much as possible net energy from both types of food as quickly as possible. In this collective foraging scenario, not only the number of active forager but also the task allocation of foraging robots on each type of food directly related to the performance of the group. Figure 1.4 shows screen shot of simulation for a group of heterogeneous robots collectively forage in a environment with two types of food: red and blue. In the experiments, a heterogeneous robot group first collective forge in the

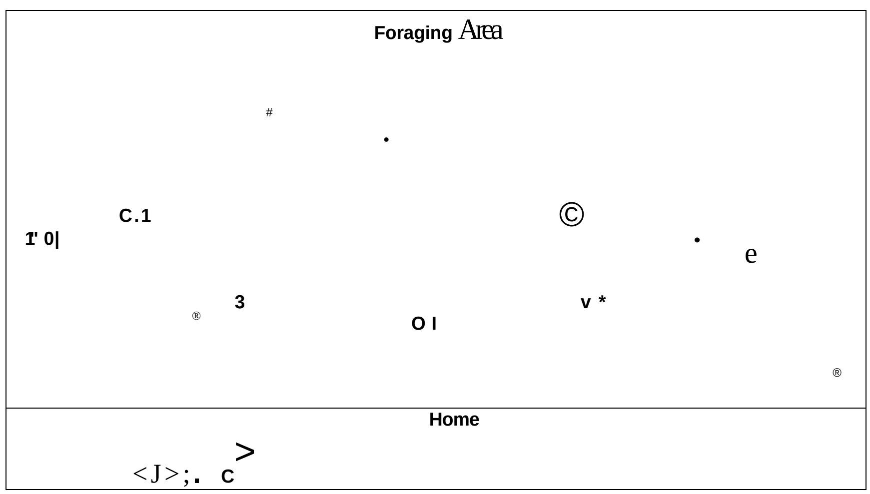


Figure 1.4 Heterogeneous Robots Collective Foraging Simulations

environments with different ratio of food from each type. Through interactions between robot and environment and between robots, group has both division of labor and task allocation at group level according to the environment and achieves highly improved group performance compare to the control group. Design group also shows flexibility and robustness in a dynamic changing environment. Figure 1.5 shows the relations between homogeneous group and heterogeneous group in this study. Two hypothesizes

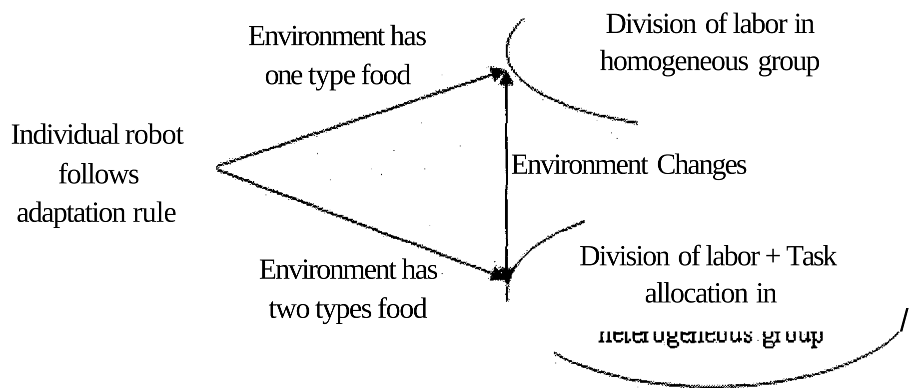


Figure 1.5 Collective Foraging Task in This Study

are proposed to stimulate the study in this thesis:

Hypothesis 1 That is possible, in a group of homogeneous robots, to use information of local sensing and communication to develop local interaction rules at the individual robot level so that the group behavior of optimal division of labor can emerged and foraging performance can be improved. (Chapter 4)

Hypothesis 2 That is possible, in a group of heterogeneous robots, to use information of local sensing and communication to develop interaction rules at the individual robot level in order to achieve desired task allocation as well as division of labor at group level. (Chapter 5)

1.2 Main Contributions

We developed a new computer simulation program to simulate multi robots collective foraging. The simulation program simulates collective foraging activity with both homogeneous and heterogeneous robot groups. Foraging response model is adapted to generate division of labor at robot group level. A set of adaptation behavior rules are designed for individual robot, based solely on local sensing and communication. Through the interactions between individual robots and foraging environment combine with interactions with other robots, robot switches the task between foraging and resting, complex group behavior of optimal division of labor achieved in a self-organized manner. Group collective foraging performance greatly improved.

The local adaptation rules have been expended and generalized for robots who foraging in a environment with two types of food. Division of labor between foraging robots and resting robots as well as task allocation among foraging robots according to different type food emerged at the group level. The group foraging performance in term of energy efficiency and time efficiency demonstrate a significantly improved. Flexibility, robustness, and scalability of the group are also demonstrated in the experiments.

Robot groups exhibit emergent group behavior of division of labor and task allocation (division of labor) in both cases. The group also exhibits the flexibility to the environmental changes (in food density) and redundancy to respond to the perturbations of individual robot foraging malfunction.

1.3 Organization of Thesis

This thesis is organized as 5 chapters: Chapter 2 is dedicated to a thorough survey of background and related work. It starts by giving a brief introduction of behavior based control structure, communication approaches used in multi robotic system design. Then the origin of the swarm robotics: self-organization and swarm intelligence are introduced. The chapter then gives the detailed reviews of typical group tasks in swarm robotics domain, focuses on discussion of the previous work done in swarm robotic system especial on collective foraging. Finally, some of the experimental tools and simulation platforms widely used in multi-robot domain is introduced.

Chapter 3 describes the methodology used throughout the thesis to support the investigation of the hypothesis. It starts from the design of behavior based control for robot model in simulation program. A detailed design of computer simulation program for multi robot collective foraging is given. A description about the collective foraging task and the state machine is presented then. Then the mechanism of division of labor used in design is explained. Finally we explained the performance metrics and measurements used in this study.

The main work for this thesis starts from Chapter 4. In this chapter, a simple adaptation mechanism is proposed for a homogeneous group robots engaged in the collective foraging task. The proposed adaptation rules are then introduced to regulate the behaviors for each robot, so that the robots can switch task between foraging and resting. A set of experiments has been used to investigate the optimal division of labor emerged at group level. Flexibility and robustness of the design group are also are demonstrated in the experiments.

In Chapter 5, we extended the adaptation local rules for a group of heterogeneous robots to forage in an environment with two type of food. Individual robot not only can switch between foraging and resting but also can switch from one type of food to the other type. A set of experiments are designed to demonstrate the optimal task allocation emerged at the group level according to the ratio of the food in the foraging area. The adaptability and flexibility of the group are also been tested.

Chapter 6 contains general conclusions, contributions from the study, and directions for future research.

CHAPTER II

BACKGROUND AND RELATED STUDIES

This chapter provides an overview on the background and topics related to the study of swarm robotics. It starts by giving a brief introduction of the behavior-based robotics in Section 2.1. Section 2.2 introduces the communication strategies were often used in multi-robotic system. Section 2.3 introduces concept of emergence and self-organization. Section 2.4 briefly introduces the origin and application of swarm intelligence. Section 2.5 introduces concept of collective robotics. Section 2.6 presents detailed reviews of typical group tasks in swarm robotics domain. Section 2.7 describes main experimental platforms and simulation tools used in swarm robotics research.

2.1 Behavior-based Control

Regardless of the intended applications, each robot in a multi-robot system requires a preprogrammed control process by which it can processes input from sensors and responds with appropriate actuations. The traditional Artificial Intelligence (AI) based approach decomposes each required task into a series of functional units. Figure 2.1 shows an AI based controller of a robot which is designed to perform a searching task. The perception and modeling unit generate an abstract representations of the outside world around it in order to plan the future actions. If the perception from sensor changes, all functional units follows it needed to be change too. Because of it's series structure, any disfunction in one unit will cause whole system disfunction. This approach is attempted to rely on building an abstract world model and then use logical inference to create new knowledge about the world. The problem for the AI based controller is that the system neither flexible to changes nor robust to function errors. It focus on intensive central data processing, large memory size and high level reasoning but not flexible to changes. On the other hand, Brooks [67] presented a behavior based

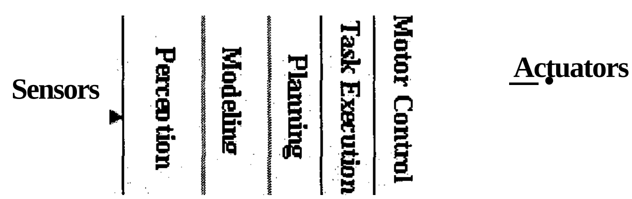


Figure 2.1 Traditional AI Based Robot Control Structure

structure to design a robot which is simpler but also flexible and robust to changes. It differed fundamentally from AI control structure by decomposing the control system based on task achieving behavior layers. Figure 2.2 is a behavior based controller for a

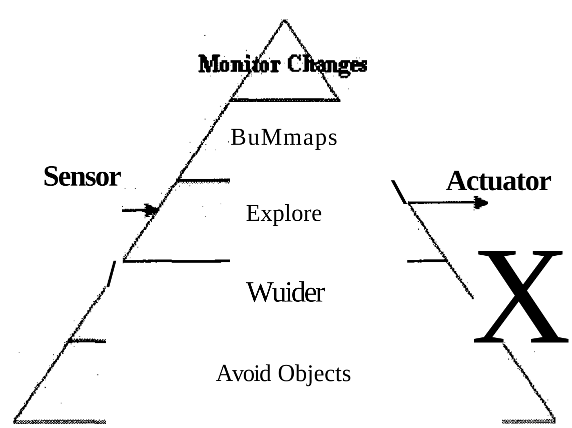


Figure 2.2 Behavior Based Control Structure

robot perform same searching task. Lower layer in a behavior based control structure accomplishes more fundamental action. More complex behavior will build on the top of basic behavior. In a search behavior, the basic layer in control structure is avoiding objects. It is also the basic behavior for any other behavior to be success. When this is achieved, the next layer is built on top of it, and the process is repeated. Because all behavior layers are parallel, any error happens in the structure only affects the layers on top of it. Most behavior-based systems are also reactive. They do not necessarily maintain a detailed model of outside world which requires massive internal memories. All the information are obtained from the input of the robot's sensors, robot simply acts in response to the stimulus it encounters, therefore the robot is able to react quickly to environmental changes and also robust to the errors. The behavior-based approaches have been widely used as the design strategies for the robots used in swarm robotic systems.

Behavior based control structure was used to design for the individual robot perform foraging task in this study. See section 3.2 for the details.

2.2 Communication Strategies

Cao [3], Arai and Pagello [5], and Dudek [6] all focus on the importance of the communication in design of a multi robotic system. Every multi robotic system requires a type of communication in order to achieve a useful coordination among robots. Here we divide the common approaches of communication into these two parts: explicit and implicit communication.

Explicit communication consists of sending and receiving of the messages that are intended to convey information. Messages can be sent from one individual to another or from one individual to many recipients, often called "broadcasting". Explicit communications become increasingly complex and impractical when number of robots in a system large.

Implicit communication, also known as stigmergy in biology, occurs through the environment. Messages are *"sent"* by altering some aspect of the environment which is then sensed by other robots. The environmental alteration can be intentional, as in the case of trail-laying of Russell [7] in ant foraging or unintentional, as in clustering or sorting tasks of Deneubourg [8], Melhuish [9]. Implicit communication occurs as a side effect of other actions, or "*through the world"*, whereas explicit communication is a specific action designed solely to convey information to other individuals. Information gained through implicit communication often tends to be limited. As the environment is available to all individuals in the system, information acquisition and processing is often much faster, but the messages cannot be designated for particular individuals. Implicit communication also scales well for potential future applications requiring hundreds or thousands of robots [10].

In swarm robotics, implicit communication often happens through the multiple interactions between robot and environment and between foraging robots.

2.3 Emergence Property and Self-organization

Self-organization was originally defined in the context of physics and chemistry to describe the emergence of macroscopic patterns out of processes and interactions defined at microscopic level. Pferfer [11] refer it as *"a process by which patterns are formed in the systems contain a large number of elements"*. Bonabeau [14] extended the concept of self-organization to describe the interactions in social insects and defined as follow: *"Self-organization is a set of dynamic mechanisms whereby structure appear at the global level of a system from interactions among its lower components. The interactions among the system's constituent units are executed on the basic of pure local information without reference to the global pattern, which is an emergent property of the system."*

Pferfer [11] pointed out three properties of the system with emergence as follows:

- A surprising property of a system that is not fully understand.
- A property of a system not contained in any one of its parts. It requires many components whose behavior is based on local rules.
- A global behavior that arises from multiple agent-environment interactions.

The interactions in the system that display emergent phenomena are non-linear which implies that the "*global behavior can not be obtained by summing the behavior of it isolated components "(*[12])

Biological scientists adopt and extend the concept to describe a wide range of collective phenomena and group behavior often observed in animals or social insects such as: geese flying in a V-shaped formation, fish traveling in schools, and aggregation behavior of bees, wasps, ants, and termites. Typically, in such systems, large populations of simple individuals interact locally with each other and with the environment. Despite the relative simplicity of the individual behaviors and decentralized control structure, complex collective behaviors and patterns which emerge from the local interactions can shown at group level. The terms *"complexity"*and *"patterns"*were used here to describe the global characteristics in such self-organized system. The relative simplicity of an individual in such system is insufficient to explain the organization, flexibility, and robustness exhibited at group level.

Recent research shows that Self-organization is a major component of a wide range of collective behavior in social insects ([13]). It usually relies on four basic ingredients ([14]): (1) positive feedback, (2) negative feedback, (3) balance of exploitation and exploration, and **(4)** multiple interactions. In these systems, positive and negative feedback of interactions among the individuals plays essential role for such phenomena. The positive feedback typically promotes the creation of structures and amplifies the fluctuations of the system. Conversely, negative feedback serves as a regulatory mechanism to counterbalance positive feedback and helps to stabilize the collective pattern.

In this study, a self organized multi-robotic system is design to achieve a highly improved collective foraging task. With each robot follows the designed interaction and adaptation rule, a coordinated group behavior of division of labor is emerged at robot group level.

2.4 Swarm Intelligence

The expression of "*swarm intelligence"*was first introduced by Beni and Wang [15] to describe the work on their cellular robotic systems. Cellular robotic systems in their work drew direct inspiration from biological organisms to generate an architecture based on decentralized hierarchies of robotic cells. Hackwood and Beni [16] gave first definition of the term swarm intelligence *:"a property of systems of non-intelligent robots exhibiting collectively intelligent behavior".* Deneubourg, Theraulaz, and Beckers [17] who study swarm intelligence from an ethological perspective, define a swarm as "a *set of (mobile) agent which are liable to communicate directly or indirectly (by acting on their local environment) with each other, and which collectively carry out a distributed problem solving".* Then Bonabeau [14] extended the definition of swarm intelligence to include artificial design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies. Here swarm intelligence is not a property of a system but instead a design approach for a distributed system.

The study of swarm intelligence takes motivation directly from the self-organized group behaviors observed in social insects [18]. Social insects often demonstrate three desired characteristics in their group behavior: robustness, flexibility and scalability. Social insects are highly robust; their self-organized systems can still work even after losing some of system components or changing the environment parameters considerably. System shows the capability to adapt to unknown or dynamic changing environment. The biological systems have this level of flexibility and can easily switch their behaviors when problems change. For instance, ants are so flexible that they can solve foraging prey retrieval and chain formation problems with the same base self-organized mechanism. Natural swarm system also has scalability to expand a self-organized mechanism to support larger or smaller numbers of individuals without impacting performance considerably.

Examples of collective behavior and pattern in social insects include herds of sheep, schools of fish, and cluster behavior of ants, as shown in Figure 2.3. In these examples, it has been shown that solutions exist that don't require direct communication but instead relying on stigmergy and self organization. Swarm intelligence also can be considered as an engineering application of study of Self-Organization (SO) in ethnology. A more recently definition of swarm intelligence by Dorigo and Birattari [19] proposed as follow: *"Swarm intelligence is the discipline that deals with natural and artificial systems composed of many individuals that coordinate using decentralized control and self-organization. In particular, the discipline focuses on the collective behaviors that result from the local interactions of the individuals with each other and with their environment".*

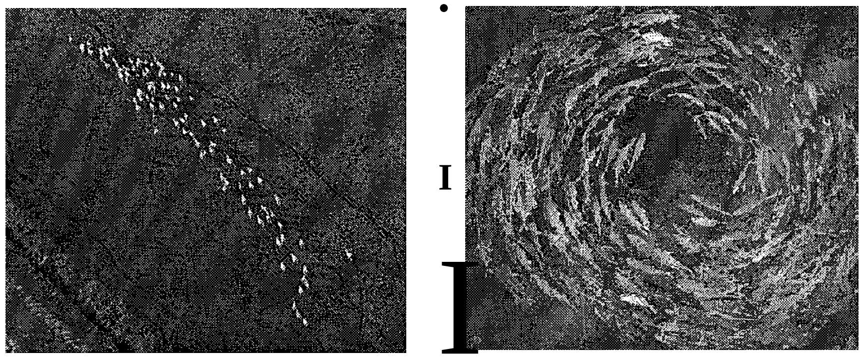


Figure 2.3 Examples of Collective Behavior. From http://www.wikipedia.org

They also pointed out that a system with swarm intelligence should have the following properties:

- A large number of individuals in the group.
- The multiple interactions among the individuals are based on simple behavioral rules. Only local information the individuals exchange directly or via the environment (stigmergy).
- The collective behavior or pattern of the group results from the multiple interactions of individuals with each other and with their environment.

There are a number of swarm intelligence based algorithms have been developed in the last decade inspired from nature swarm intelligence systems:

1. **Ant colony ptimization** is inspired from the foraging behavior of ants in finding the shortest paths from a food source to their home ([20, 21, 23]). It was first applied to the travelling salesman problem (TSP), and has been recently applied

to many combinatorial optimization problems to find approximate solutions to difficult optimization problems.

2. **Particle Swarm Optimization** (PSO) is another example of successful swarm intelligence based design([92]). It is inspired by the group behaviors in flocks of birds and schools of fish, as shown in Figure 2.4. In PSO, the individual is typically modeled by particles in a n-dimensional space that have a position and a velocity. These individuals fly through the search space and update their position and velocity based on their own experience and the experience of their neighbors. PSO is a global optimization algorithm for dealing with continuous optimization problems and has been applied to a variety of applications, such as the training of artificial neural networks and for finite element updating.

2.5 Collective Robotics

Collective robotics focuses on the study of a multi-robot system which is designed to cooperatively achieve a task that is beyond the scope of a single robot. Swarm intelligence based approach often used in the design of such robotic system in order to accomplish a desired group behavior. In a collective robotic system, multi-robots cooperation has become increasingly important for a number of reasons. Among these reasons, Cao [3] points the following:

(a) Tasks are often inherently too complex for a single robot to accomplish.

(b) Several simple robots can be a cheaper and easier solution than one powerful robot.

(c) Multi-robot systems are generally more flexible and fault-tolerant than single robots acting alone.

Parker [4] identifies eight primary research topics in the field of collective robotics: (1) Biological inspiration (2) Communication architectures (3) Localization, mapping and exploration (4) Cooperative object transport (5) Motion coordination (6) Reconfiguration robots (7) Learning

The field of collective robotics naturally divides two different directions. The first primarily concern with morphology and considers robotic modules physically attached together which self organize to generate global movement. It is done by changing the configuration of the attachment of the robots referred to as self-reconfigurable robots.

The second direction studies the multiple interactions of autonomous robots co-operating in the execution of a task. Such works relies on swarm intelligence based design. It also called swarm robotics. Collective foraging, collective transportation and distributed movement in formation are several examples.

Our study present here belongs to the second direction of collective robotics. A group of simple robots collectively forage in a foraging environment, through multiple interactions between robot and environment and between foraging robots, a desired collective behavior "division of labor"emerge at robot group level to help group achieve highly improved group foraging performance.

2.6 Swarm Robotics

Swarm robotics is an application of swarm intelligence based algorithm in designing the control system for a multirobotic system. It is an approach which inspired by the collective behavior in social insects and the study of self-organization system. Sahin [24] proposed a definition for swarm robotics:

"Swarm robotics is the study of how a large number of relatively simple physically embodied agents can be designed such that a desired collective behavior emerges from the local interactions among the agents and between the agents and the environment."

Compare to general multi-robotic systems, swarm robotic system usually includes a large number of simple robots with distributed control structure. Simple robots in a swarm robotic system have only limited sensing, communication and computation abilities. The required group behavior can not accomplished by single robot due to the relative incapability or inefficiency of the robots. At group level, system shows flexibility, robustness and scalability while performing the desired task.

Since robots are simple and global information about required task is not available to individual robots, desired group behavior only rely on the local behavior of robot. The biggest challenge in designing such system is to determine the local behavior rules for individual robot so that desired collective behavior can be achieved through the multiple interactions between the robots and between the robots and the environment. Studies of collective behavior in biological systems show some hint for the design of swarm robotics. For example, the flocking behaviors in a group of birds or fish, the aggregation patterns in insect societies [50] suggest that the complex group behavior could emerge from the interaction from the simple individuals in a self-organized manner.

During last 20 years, a large number of research works have been done in the domain of swarm robotics. Bayindir [26] presented a review of studies for swarm robotics from different research axes. They classified existing studies according to the axes of modeling, behavior design, communication, analytical studies and problems. In this section, the studies in swarm robotics are categorized according to the group task that the systems aim to undertake, including collective foraging, aggregation and flocking, collective clustering, collective searching, cooperative transport.

Collective Foraging

Among the tasks which performed by a multi-robotic system, collective foraging is probably most well-studied scenario due to its strong biological basis.The main biological inspiration comes from the observation of foraging activity in ant or bee colonies. Although most of researches in swarm robotics have only been done in research labs, potentially, there are many actual real-world applications for foraging robots, for instance cleaning, harvesting, search and rescue, land-mine clearance or planetary exploration [49].

The robots engaged in a foraging activity need to search for the food in a foraging environment, once food found, robots collect food back to a specific location, generally named home or nest. In most cases, the food can be handled by a single robot, therefore one robot may be able to accomplish the foraging task if given enough time, but a group of robots working together may be able to achieve much higher foraging efficiency especial when robots foraging in a large foraging space.

However, due to the limitation of foraging space and competition between foraging robots, efficiency of group foraging will reduce instead of increase if the size of the group keeps increasing. For example, in an area full of robots, clearly no robots can be successful as they are trying to avoid each other all the time. In this case, all the energy was use for avoiding other foraging robots and the time to finish the task could be infinite, i.e. efficiency of the group is very low. Besides the spatial interference, competition for the food resource has a negative impact on the efficiency of the group as well. In particular, a limited availability of food resource results in wasted energy of the group, as the activity engaged in the foraging task will continue consume energy. Taking time and energy efficiency into account, there is an optimal number of foraging robots in order to achieve optimal group foraging efficiency. The main challenges in collective foraging task are (1) Availability of food in foraging environment is unknown. (2) Environment is changing. Especially for robots with only limited sensing, computation and communication abilities.

Early work in robot foraging focused on the design the behaviors of robot to fulfill the foraging task.

Matari'c implemented foraging experiment using a group of real robot ([28, 30]). Behavior based control structure was used for each robot. Foraging behavior is achieved by combining a set of basic interactions, including avoiding, dispersing, searching, homing and resting, which are called behavior primitives.

In Mataric's [30] robot foraging experiments, they prove that interference among robots in a bounded environment plays a negative role on the performance of the group. To improve efficiency, several different mechanisms have been used to reduce the negative effect of the interference. Lerman [40] developed a macroscopic probabilistic model to quantitatively analyze the effect of swarm size and interference among robots on the overall performance based on Mataric's experiments. They found the group performance improves as the system size grows at first, but declines for larger group sizes due to the effects of interference.

Fontan and Mataric [38] applied a spatial division approach to minimize the interference for robots near the home region, where competition for space occurs much more frequently. In their experiment each robot is allocated a specific working area and delivery region so that they are less likely to collide with each other in a relatively crowded area.

Similarly, Ostergaard [39] introduced a bucket brigading strategy to overcome the negative impact due to the high density of robots in the environment. The robots hand off a resource from one robot to the next, but no explicit information is given about where the robots should position themselves in the brigade. Instead, robots drop food whenever they encounter with a robot without food.

Matari'c also explored the reinforcement learning [31] approach in designing the foraging behavior in a group of physical robots [32], Heterogeneous reinforcement function and progress estimators are applied to overcome the poor performance of traditional reinforcement learning methods in physical robots.

Balch [33] built a group of robots for a trash-collecting task. Robots used miniature color cameras for discriminating between various object classes: trash, wastebaskets and robots. Cooperation among robots is implemented implicitly using only inter-robot repulsion, which is the combination effect from the behavior "look for trash"and "avoid static obstacles ". Robots accomplish the task without using any explicit communication.

Dario Floreano and coworkers [34] use behavior-based robotic design to implement the controllers for their robots, named evolutionary robotics. There are two projects related to foraging: (a)Evolution of Division of Labour in Social Insects and (b) Evolution of Cooperation in Artificial Ants. In their studies, two types of items need to be collected: the small items which can be handled by a single robot, and the medium items which need two robots to push cooperatively. They investigated the role of relatedness, levels of selection and group size on evolution of cooperation and labor division in ant colonies [36, 37].

Sugawara [41] proposed a virtual pheromone system in their robot foraging task. In their experiments, chemical signals are simulated with the graphics projected on the floor, and the robots decide their action depending on the color information of the graphics. A robot leaves a pheromone when it moves home carrying food, so the robots without food can detect the pheromone and follow the trail to get food, which is similar to the foraging behavior observed in ant colony.

In order to regulate number of foraging robots in foraging environment that group can achieve optimal foraging performance, there has been increased work in investigating the mechanism of division of labor in the robot group. Two kind of labor are considered here: actively foraging and resting at home. A threshold based approach inspired from biological systems, as first described by Theraulaz et al. [42] in investigating the division of labor in social insects, is mostly used to design the task allocation mechanism.

Krieger [43, 44] implemented a group with 12 real robots engaged in a foraging task. Each robot in their experiments is characterized with a different randomly chosen activation-threshold in order to regulate the activity of the team. For the robots in the nest, once the energy level of the nest drops below its activation threshold, the robot is triggered to go and collect food-items. With this simple mechanism, their robots demonstrate self-organized task allocation efficiently in decentralized way.

In stead of purely threshold-based approaches, Jones and Mataric [47] designed an adaptive method for task allocation between collection of red or green pucks. In their experiment, robots do not communicate but observe each other and maintain a limited memory of observation in order to decide which task needs to be performed.

Labella [45, 46] introduced a simple adaptive mechanism to change the probability of individual robot in the group. Probability of leaving home for one robot is adjusted dynamically based on successful retrieval of food. They reward successful food retrieval and punish failure in order to vary the probability of robot leaving home. With difference of foraging probabilities among robots, division of labor emerged at group level. They showed that the efficiency of the group can be improved with this simple adaptation rule. They also Concluded that complex group behavior can be formed without explicit communication.

Guerrero and Oliver [48] present an auction-like task allocation model, partially inspired by auction and threshold-based methods, to try to determine the optimal number of robots needed for foraging, however, the demands of communication between robots during the auction process constrains the scalability of their method.

Although there are a few work targeting effective collective foraging, for instance, Krieger [43], Jones and Mataric [47] and Labella [46] as discussed above, they are more interested in how division of labor can be achieved using different strategies, mostly based on a threshold-hold approach, while pay little attention on the effect of interactions between foraging robots on the performance of the group. On one side, the interference or competition arisen from interference between robots may play negative roles on the performance of the group, on the other side, the interactions between the robots could also provide the group information about the environment or the foraging process itself, for example, the foraging states of other foraging robots can help to estimate the foods available in the foraging area.

To address these problems, the studies of this thesis will be mostly based on their work by taking a threshold-based approach to achieve an optimal division of labor in performing collective foraging, while more effort will be focused on dealing with design of the interaction rules between foraging robots in order to regulate the behavior of the robots in the group, and also will extend the interaction and adaptation rules to a heterogeneous robot group collective forage in an environment with two types of food.

Aggregation and Flocking

Aggregation and flocking behavior in group animals or insects was often observed in the natural world. Aggregation helps animals or insects to avoid predators, resist hostile environmental conditions and find mates [50]. Figure 2.4 shows aggregation of a swarm of insects. The aggregation behavior is often facilitated by environmental clues, for example, light or sounds. Aggregation is also at the basis of the emergence of various forms of more complex group behavior, for example, flocking. Flocking is the aggregate motions of a flock of animals, for example, flock of birds, a herd of land animals, or a school of fish. Figure 2.5 shows a group of birds conducting flocking behavior in flight.

In the swarm robotics domain, the aggregation task normally refers to gathering a group of randomly scattered robots in the environment to form a robot cluster. One of the early robotics applications of aggregation behavior is by Kube and Zhang [51]. In

Figure 2.4 Insects Aggregation. From http://www.wikipedia.org

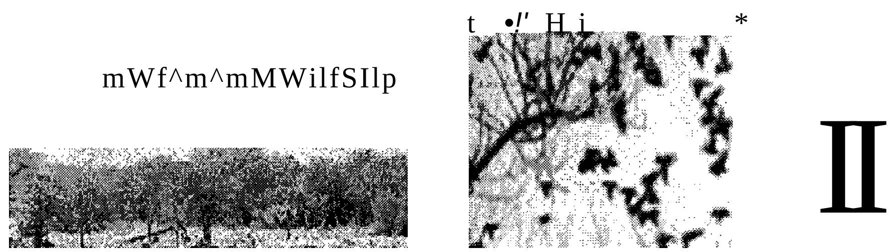


Figure 2.5 Birds Flocking. From http://www.wikipedia.org

their experiment, the robots are attracted to a lighted box first. Once the robot reaches the box, they block the light from the view of other robots, resulting in no more robots being attracted to the box when there are enough robots blocking the light completely. The required group size has to be equal to the number of robots needed to block the light.

Melhuish [52] implemented an aggregation behavior with a group of robots in a different way. In their experiment, robots are required to form clusters of predefined size around infrared beacons. Each robot in the group is able to produce sound simultaneously which may generate synchronized choruses. A robot can estimate the size of the aggregation by the energy it sense. The size of the cluster of robots can be controlled when an individual robot is designed to approach a group which is below some required size, and leave a group which is above that size.

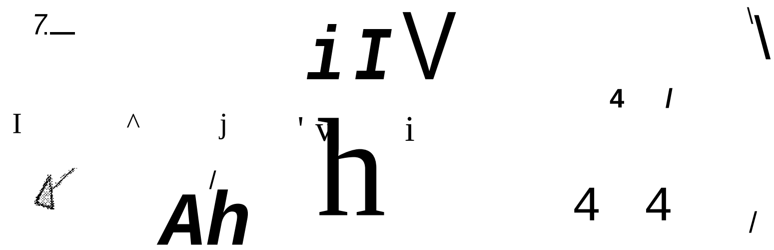
Trianni([53, 54]) used genetic algorithms to evolve neural network controllers for a group of robots in an aggregation task. The genotype specifies the connection weights of a simple perception, and the fitness function of the evolution is defined as the average distance of the robot group from its centre of mass for each epoch. They observed that two types of aggregation strategies emerge from the evolution, including a static cluster where robots create a very compact and stable aggregation without changing their relative position, and a dynamic cluster where robots form a looser cluster but move as a group. The dynamic cluster behavior is believed to be more scalable than the static. Similarly, Bahgec'i and Sahin [55] tried to achieve the aggregation behavior by evolving a neural network.

Soysal and Sahin [56] presented a probabilistic aggregation strategy for a swarm of s-bots. The aggregation behavior is achieved by combining four basic behaviors: obstacle avoidance, approach, repel, and wait. Robots use sound sensors to approach or repel from the cluster. The transitions from state repel to approach, and from state wait to repel are triggered by two different probabilities. They designed four different control strategies by varying these two probabilities. Experiment results showed that three different group behaviors emerge from the interactions among the swarm: segregation like behavior, static clustering, and dynamic clustering.

Gamier [57] tried to reproduce the aggregation behavior observed in cockroaches using Alice micro-robots. The aggregation process for the robots is directly inspired by a biological model of displacement and aggregation developed from experiments with first instar larvae of the German cockroach Blattella Germania (Jeanson [58]). A group of 10 or 20 robots are put in a circular area for a certain period of time. Although each robot has only limited perception and communication abilities, with a small set of simple behavioral rules, the results showed that the group of robots is able to select collectively an aggregation site among two identical or different shelters.

Flocking is a classic task in swarm robotics refers to a group of robots move together with coherence and flexibility. Reynolds [59] first study the flocking behavior with his simulation program, Boids. His approach assumes a flock is simply the result of the interaction between the behaviors of individual birds. The model is based on simulating the behavior of each bird independently. Working independently, each agent try both to stick together and avoid collisions with one another and with other objects in their environment. There designed rules which individual has to follow, as shown in Figure 2.6.

- Separation to avoid collisions with nearby neighbors
- Alignment to steer towards the average heading of neighbors
- Cohesion to steer towards the average position of neighbors



(a) Separation (b) Aiipmatl •i-iCck^wu

Figure 2.6 The Rules Applied in Boids Flocking Simulation. From Reynolds [59]

With these three simple rules, the flock moves in an extremely realistic way with complex group motion like we observe in nature. Following Reynolds' work, several other computer models have been proposed, leading to the creation of a new area in computer graphics ([60]). At the same time, theoretical analyses of the stability of flocking behaviors have also been presented by several researchers in the area of statistical physics and complexity theory([61, 62, 63, 64, 65]).

Kelly and Keating [66] demonstrated flocking behavior using a group of 5 mobile robots. Each robot using a behavior based design ([67]) with four rules: (1) avoid objects (2) if no other robots are visible become a leader and wander, (3) if in a flock try to maintain position, (4) if a flock can be seen in the distance, speed up and head towards it. Group is robust to the individual failure because the selection of leader is dynamic.

Hanada [69] presented adaptive flocking for a group of robots to navigate automatically in an environment with obstacles. Robots only have limited sensing ability, no memory, no leader and no communication with each other. In spite of these limitations, given any arbitrary initial positions, a large-scale swarm of robots is designed to navigate toward a goal position while locally interacting with other robots in close proximity. According to environmental conditions, each robot interacts with two neighboring robots within its sensing range by maintaining a uniform distance so that the group can be split into multiple groups or re-united into one. They showed in an experiment that the adaptive flocking algorithm enables the robots to navigate toward achieving a mission while adapting to an unknown environment using only local interactions.

Vaughan [68] demonstrated a robot system that achieves a sheepdog-like task, gathering and fetching a flock of duck to a pre-defined goal position. The rule for robot is (1) attracted to each duck-let with a magnitude proportional to their mutual distance. This force causes the robot to move towards the dock. A second force (2) repels the robot from each duck with a magnitude proportional to the inverse square of their mutual distance. This prevents collisions. A further force (3) repels the robot from the goal position with a constant magnitude. This is the first automatic system to exploit an animal's behavior to achieve a useful task.

In the above designs, robots need to be able to sense the position information of their neighbors in order to achieve the flocking behavior. Alternatively, Nembrini [70] implemented an emergent flocking behavior using only local communications, without requiring the position information at all. Each robot is equipped with a range-limited communication, which can be used to calculate the number of peers within range and exchange very limited information between nearby neighbors when required and controlled by three simple rules: (1) avoiding if too close, (2) turning back to recover the connection, (3) making a random turn, to maintain the aggregation. The behaviors for "turning back"and "random turn"are determined by the number of neighbors within communication range. In their experiments, a spot light source is used to attract the swarm to flock around. But the robots do not need to know the direction and position of light source at all. Instead, the robots are categorized to be illuminated and non-illuminated ones, depending on whether the robot can detect the light source or not using Omni directional light beacon sensor. The emergent flocking behavior is achieved with an extra rule that 'turning back' has highest priority whenever the robot lost illuminated neighbors. With the same algorithm, Bjerknes [71]developed an emergent swarm taxi behavior by introducing differential avoiding distance between the illuminated and non-illuminated robots.

Collective Searching

Deploying a group of simple robots for searching can significant overcome the lack of sensing abilities of a single robot, especially in a large space environment. The early work in this research area has considered only a small group of robots and focused on complex inter-robot communication and motion planning, for example, Parker [79], Jennings [80]. However, there are increased works aiming to coordinate a large group of robots through simple individual behavior rules.

Reif and Wang [81] introduced an approach, which uses the social potential field, to control a very large robotic system. They defined a simple artificial inverse-power force law between pairs of robots or robot groups to coordinate the motions among robots. The robots use only distance information to their immediate neighbors to generate control rules: whether attracting to or repelling from other robots depends on the distance between each other. In their study, they have demonstrated several group behaviors in simulation for the robots using different force laws with social potential fields, for example, static and dynamic clustering, guarding and escorting.

Dudenhoeffer and Jones [82] extended the social potential field model by introducing a neutral zone within which other behaviors may exhibit themselves. The neutral zone permits the wandering behavior to activate and promotes expansion of the collective in a specific direction. Their simulation results showed that the group can achieve a robust coordinated behavior in the presence of agent death (failure) and sensor imperfections in a simulated environment.

Furthermore, Dudenhoeffer and coworkers ([83, 84, 85]) have implemented the social potential fields model with a group of real robots for a spill finding task, in which the robots need to search through corridors and rooms to locate a spill, and then cooperatively form a perimeter around a chemical spill. In their study, onboard adaptation techniques are also introduced to increase the diversity of the group, which helps combat the problems of redundancy during searches.

Hayes [86] introduced a collaborative exploration strategy for a group of robots searching for an infrared beacon placed in a corner of the exploration area. The robots use simple binary signaling to attract surrounding robots. Their results showed that such a simple collaborative mechanism could drastically improve the team performance by reducing the arrival time of the group as well as the group power consumption. They later introduced the same communication mechanism for a distributed odor localization task ([87, 88]) and observed a significant improvement over the strategies without collaboration.

Correll and Martinoli [103] evaluated a jet turbine inspection using the real miniature robots Alice II ([90]). Each robot searches for blades randomly, and circles the blade found from the contact point and leaves from the blade's tip. There is one parameter to control the duration that the robot will follow the blade contour. They investigated the overall collective behavior of the swarm using probabilistic macroscopic models. The results show that the time for completing the inspection decreases sublinearly with the number of robots, and that the inspection redundancy increases with the swarm size if robots prematurely leave the blade they are inspecting.

Pugh and Martinoli [91] presented a multi-robot search algorithm inspired by Particle Swarm Optimization (PSO). In order to use the basic PSO ([92]Equations to determine the desired velocities, robots are required to communicate among themselves and have perfect knowledge about their location in the environment. They made some modifications to the basic PSO algorithm due to the limitations and characteristics of the robot system, for instance, movement limitations and physical size of robots. The simulation results showed that the performance can be improved by increasing either the number of robots or the communication range. Another modified PSO-inspired searching algorithm without global positioning has also been investigated in their study. The simulation results suggested that using no global positioning but storing the best locations may increase performance as less clustering will occur in poor regions of the search space.

Collective Clustering

Collective clustering is also considered as one of the classic group task for swarm robotics [14]. In a nest of ants, the larvae, cocoons and eggs are arranged as separate clusters in entirely different parts of the nest. Deneubourg [72] proposed a simple model to explain these phenomena. The general idea is that ants pick up and drop objects at some other location where more objects of that type are presented. The probability for picking up or dropping down an object is modulated as a function of an ant's recent observation, that is, how many of the same objects it has met in the recent past. Their model captures many features of the ant sorting behavior.

The first implementation of collective clustering using a group of robots is presented by Beckers [73]. The task for their robots is to gather 81 randomly distributed objects and cluster them into one pile. Each robot has three simple behavior rules: (1) wandering, (2) obstacle-avoidance, and (3) puck-dropping. The puck-dropping behavior is triggered by the micro switch whenever the robot bumps into a cluster of three or more pucks. There is no explicit communication (IR or radio link) between the robots. The robots only react to the local configuration of the environment. Although the pucks are clustered into several piles at an early stage, surprisingly, their experiments showed that it can always form a single large cluster after a long duration. They also studied the performance of the group of robots as a function of group size and found that the optimal number of robots is three, with respect to the experimental configuration in the lab.

Martinoli [74] have carried out more experiments using the same algorithm. They used a group of Khepera robots with sophisticated grippers, able to grasp and lift the pucks. Experiments were performed with one to five robots, and 20 pucks initially randomly scattered in an 80X80 cm^2 arena. They found that the pucks clustered into several piles, instead of a big cluster as observed in Beckers [73], and the size does not vary much after 80 minutes.

Melhuish and coworkers continued to extend the collective sorting from two types of objects to multiple types of objects. An extra sensor has been added to the robots to distinguish different types of object. In Melhuish [75], simulations test the refined sorting mechanism. They defined four behavior rules for the simulated robots. Robots drop the puck only when it is the same type as the pucks in front of them. They tested the sorting mechanism using 1 to 20 different colors of objects in simulation, and found that a high level of clustering can be achieved using small number of types of colored objects, for example, 1 to 7, but has less success when the number of colors was increased beyond 10.

Wilson [76] made a further extension based on Melhuish's algorithm. They presented three different annular sort mechanisms: (1) object clustering using objects of different size, (2) extended differential pullback, (3) combined leaky integrator, and tested these sort mechanisms both in simulation and real robot experiments. They found that using the differential pullback mechanism with a large number of object type results in a lack of compactness, but a combined leaky integrator mechanism using appropriate parameters, where pullback distances are adaptive, can improve the compactness.

Wang and Zhang [77] proposed a control algorithm for collective sorting which is similar to Melhuish [75], but depends on more sensing. The difference lies in the strategy for picking up the encountered object. In their study, each robot has a wider sensing ability and can detect more than one object at the same time. The robot will pick up the object unless it is different from other detected objects. With the simulation results, they claimed that the sorting mechanism achieved a significant improvement in performance with respect to the convergence of the sorting process.

Verret [78] implemented the same algorithm with a set of RoboCup soccer robots and demonstrated the collective sorting successfully. They also studied the effect of the sensing range on the performance of the sorting algorithm, and found there was a pronounced improvement when the sensing range increased from 50cm to 100cm and shared a small further improvement when the range kept increasing.

Cooperative Transport/Handling

Transporting or manipulating a large heavy object may require the cooperation of two robots. A box pushing task has been widely used to study approaches for achieving cooperation among distributed controlled robots. Most researches focused on a fixed group of tightly coupled robots in objects transportation tasks, where robots coordinated to accomplish the task with explicit communication or without explicit communication but formed into a predefined formation.

Stilwell and Bay [93] developed a material transportation system using a group of ant-like robots. A Leader-follower mechanism is used to coordinate the motion of the group. Robots used the local force information from a top-mounted force sensor to derive the control parameters.

Parker [94] introduced the ALLIANCE architecture for their box pushing experiments. They demonstrated that the task can be accomplished even though one of two robots has failed.

Yamada and Saito [95] proposed an adaptive action selection approach without explicit communication for multi-robot box pushing. A formation behavior is executed first to form a line to touch with the box before pushing. They claimed their approach has the ability to adapt to a dynamic environment in terms of existence of other robots and task difficulty.

Other researchers contributed the research on collective transportation inspired from biological paradigm, particularly from ant colonies. Kube and Bonabeau [96] described a box-pushing experiment using a group of homogeneous robots. In their study, five robots collectively push a large box towards an arbitrary goal position indicated with a light source. Robots change position and alignment until the box can be moved (so called stagnation recovery), which is similar to the phenomena observed in ant prey retrieval. There is no explicit communication or cooperation mechanism between robots and robots do not detect the presence of other robots. However, implicit communication among the robots takes place through modification of the environment (here the box and the physical space of the environment ([73]). It is believed that such implicit communication provides the main coordinating mechanism in group transport by their robots. Although the robotic system doesn't appear to be very efficient, it is the first experiment in collective transport closest to the cooperative transport seen in biology.

Another interesting work on cooperative transport has been done by Macro Dorigo and coworkers ([97, 98, 99]) within the Swarm-Bot project. Their robots can not only push the object but also pull it towards a target location using grippers. Moreover, robots can attach themselves to each other and self-assemble into structures connected to the prey. A neural network controller shaped by artificial evolution is used to control the single robot (called s-bot). They showed that the controllers evolved for a relatively small group can be applied to larger groups to cooperatively move objects with different shape and weight. They observed that the performance decreases with group size due to the difficulty in self-assembly for a high density of robots.

Apart from cooperative transportation for single objects, some other researchers focus on efficiency of handling multiple big/heavy objects using a swarm of robots. One such study is the stick-pulling experiment of Ijspeert [100], as shown in Figure 2.7. The stick-pulling task requires collaboration between at least two robots to be successfully completed, for instance, cooperatively pulling sticks out of the ground. In their studies, they have investigated how collaboration in a group of simple reactive robots can be obtained through the exploitation of local interactions, and how to optimize software parameters of the robots' controllers to improve the collaboration rate. Microscopic and macroscopic probabilistic models are also explored to predict group behavior, and therefore help the design of individual controllers.

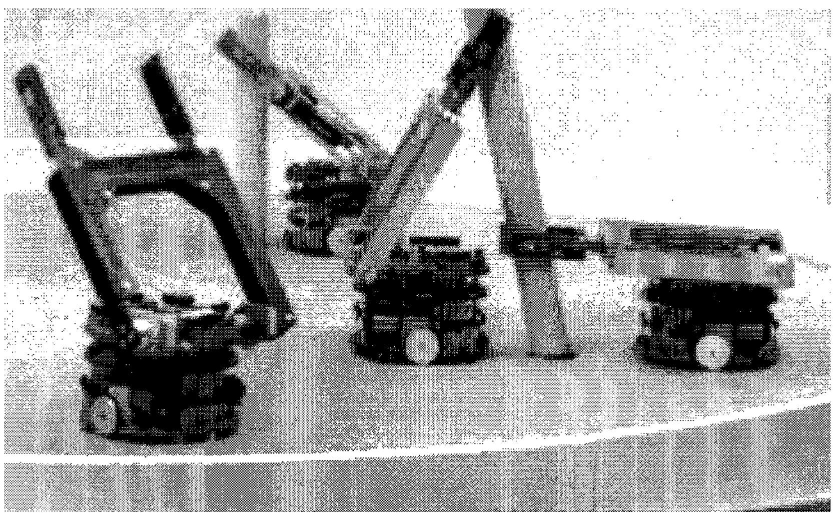


Figure 2.7 Stick Pulling Experiment. From Ijspeert [100]

2.7 Experimental Platforms

Although experiments using real robot provide realistic results for robotic study, it is often very useful to perform simulations prior to investigations with real robots. Firstly, building up models for robot and setting up experiments is much easier, less expensive and faster in simulation than with real robots. Especially for swarm robotics which normally requires a large number of robots to exhibit a collective behavior. It may take months to build and debug each piece of hardware required for the experiments. Secondly, simulation often runs much faster than real robots. Some computationally expensive algorithms, such as evolutionary algorithms, would take days or months to run on real robots. In many cases, it is impossible to work with real robots due to lack of continuous battery support or the requirement of hardware maintenance. In recent years, a number of simulation tools have become available for research in the swarm robotics domain. For example, the U-Bots and LinuxBots in the Bristol Robotics Laboratory, the Alice in EPFL and more recently the European Swarm-bot project. Martinoli and coworkers have studied the collective behavior using using the simulated Khepera and Alice robots in collective clustering ([101, 102]), collaborative stick-pulling ([100], collective turbine inspection ([103]), and self-organized aggregation ([104]).

Like most research work in swarm robotics, this study is involve with a large number of robots perform collective foraging tasks, setting up a good experimental platform plays an important role in validating the algorithms. We have developed our own multi robot collective foraging simulation program. Detailed design information about simulation program is in chapter 3.

CHAPTER III

METHODOLOGY

In the study of a swarm robotic system, individual robot in the group is typical sample in term of sense, computation and communication ability, but the group behavior which emerge from multiple interactions among robots and between robot and environment could be very coordinated and complex. A typical methodology in swarm robotics is to start from the desired group behavior, then to design the local behavior rules for individual robot to follow. In the past, there is few analysis tools are available for researchers. Even real robot experiments are needed to confirm the results from the simulations, experiments with a large group of robots still is very time consuming and costly. Recently, a complementary research tool, probabilistic mathematical modeling offers an alternative way to swarm robotics filed [110, 111]. The basic idea in the probabilistic mathematical model is to consider the interactions among robots as probabilistic events and producing either macro or micro level update equations for robot group. However, the limitation of using probabilistic mathematical model in a heterogeneous robot group remains to be solved. As the result, most studies in swarm robotics rely on the following process: the early design done through help of the computer simulations which presents limited degree of accuracy, then the confirmation of the results is sought by experiments with real robot group.

This study follows the methodology in swarm robotics. We start from the goal of collective foraging activity and from that, the desired robot group behavior is given in Section 3.1. Section 3.2 introduces the design of behavior based controller for individual robot. A computer simulation program which was used to simulate multi robots collective foraging is developed for this study. All class modules in the simulation program are introduced in Section 3.3. Section 3.4 introduces a state machine which is use to show each state in foraging activity of a robot. A mechanism which is used to achieve division of labor between robots is introduced in Section 3.5. Section 3.6 introduces the metrics were designed to measure group foraging performance in this study. Finally, measurements of the data is defined in section 3.7.

3.1 Desired Group Behavior

The main biological inspiration of collective foraging activity in this study comes from the observation of foraging activity in ant or bee colonies. Ants need to search the environment for food in order to survive and to provide energy to the colony. But not all the ants in colony can be forager at the same time: if all ants were foraging, none could defend the nest or feed the brood. If there are too many ants foraging in the filed, it is also possible that they interference each other that few ants could not retrieve food back successfully.

In study of collective foraging with a group of robots, robots acquire energy from the food they retrieve back home also consume energy while they moving and searching in foraging environment. Energy gain is only from food foraging robots retrieve back.

Energy consumption however comes from activity of foraging robots such as moving, avoiding each other. Net energy of the group is the food energy gain less the energy consumed by foraging robots. The goal for the robot group collective foraging activity is to acquire net energy from food source in environment as fast as possible.

Energy gain from collective foraging only depends on the food foraging robots retrieved. But energy consumed during foraging is hard to estimate accurately. Foraging in different environments also could leads to the difference on the energy consumption. For example, environment full of robots could cost more energy because of the interference among robots. Both energy gain and energy cost directly relate to the number of foraging robots. In this study, we simplify the energy cost as the function of duty time (the time foraging robots spend on searching and retrieving) and number of interferences between foraging robots. Interference happens when two foraging robots are close within a certain distance, both of them has to stop in order to avoid the collision, backward and turn to a random direction before reset to their previous states. In order to calculate the net energy from a foraging activity, we assume that:

- A foraging robot consumes energy at A units per second while searching or retrieving, and consumes energy at $\boldsymbol{B}$ units per second when interferes with another foraging robot. Here we assume in a given environment, number of interferences $fi(x)$ is a function of the number of active foraging robots x in the area.

- A food is retrieved by a robot will provide C units of energy back to the group.

- In a given environment, average foraging duty time of a robot (the time a robot spends on searching or retrieving) also is a function of *x*, denoted by *t*, say: $t=f_2(x)$.

Econsumed is the energy consumed per second in a foraging activity and $E_{retr}ievai$ is the energy acquired per second from a food retrieved by a robot, then we have:

$$\hat{}consumed, \qquad E_{re}f_{r}i_{eva}i \qquad \bullet jX$$

Thus the average net group energy per second for the group is:

$$E_{n}et = E_{re}i_{r}j_{eva}i \quad E_{consume(}i = \frac{C}{J2\{X\}} \cdot x \quad A(x') \quad Bf\backslash(x) \qquad (3.1)$$

Equation (3.1) shows that in order to maximize E_{net}, number of foraging robot *x* need to be optimized. Increasing x could increase *Cx* which means more foraging robots bring more food energy back home. However, more foraging robots engaged in a bounded foraging environment, more likely foraging robots have to compete for a limited resource of food so that it takes longer time for robots to find and retrieve food. Having more foraging robots could also results in more interferences *fi(x)* which decrease E_{net}. Therefore, in a given environment, there is an optimal value, say X*, for *x* that E_{net} in Equation(3.1) can be maximized. Figure 3.1 illustrates the relation between E_{net} and *x*. An optimal number of foragers X* (to gain a maximized E_{net}) lies somewhere between two extreme cases: **(a)** few robots engage in foraging in which E_{net} is small because $E_{retrieva}i$ is small and **(b)** an environment is full of foraging robots, in which few robots can retrieve any foods successfully, but energy consume on the

Figure 3.1 Net Energy from Collective Foraging vs. Number of Foraging Robots

interferences between robots continue increase so that E_{net} continue to decrease below zero. The exactly shape of the curve depends on the particular environment and robot group but the general shape of the graph should be similar. It shows that in a given environment, there is an optimal value of foraging robot number X* so that E_{net} can be maximized. If the foraging environment changes, curve will change and the optimal value of X* changes meanwhile. So the desired group behavior of a group robot on a collective foraging activity is to have an optimal number of foraging robots so that net energy of the group could be maximized.

The challenges are firstly, all robots in the group are simple robots with only local sense and communication ability. None of robot could know the availability of food in the environment. Secondly, the functions of *fi(x)* and /$_2$(.x) in Equation (3.1) are quite complex and hard to model because of the complexity and dynamic of the interactions among robots and unknown of food availability in the environment. The exactly curve of E_{net} is therefore hard to learn. Even a mathematical model can be obtained based on complex analysis, the dynamic natural of the problem and potential uncertain of the environmental conditions makes the analysis process much more complicated.

Therefore, it seems more practical to use a process which starts from the design of the interaction and adaptation bahavior rule for individual robots (a typical characteristic of the swarm robotics methodology), resulting in a desired group behavior that the number of active forager is able to adapt according to the environment. We will discuss such an approach in the following sections.

3.2 Robot Controller Design

We first designed a controller for the robot class module in the simulation program to perform the foraging task. Behavior based design which includes a system of layered levels of competence were used to design the robot controller. It is constructed in a sequence from the simplest behavior to most complex behavior. Each competence level forms a complete and fully-functional system of control layer. In designing a controller for a foraging robot, three levels of competence are designed within behavior based control architecture, Levels 1, 2, and 3. These layers are stacked in parallel, with each higher level subsuming the lower levels. In order to accomplish the foraging behavior, three basis behaviors of a robot were identified:

- Level 1: Avoiding - Collision detection with other foraging robots or boundary of foraging environment and response.
- Level 2: Searching - Search randomly for food in foraging environment.

• Level 3: Homing - Pick up a food, move back home, and drop the food.

Appropriate implementation of these basic behaviors makes it is possible for a robot to accomplish the foraging behavior. During the program development, a sequential "*layered approach"* was employed to develop each competence level. The first competence level was first written, debugged, and tested to flawless operation before adding the second level. This process was repeated until the robot achieved *"Level 3 competence"*. The basis behaviors are arranged below according their respective competence level: The level 1 behavior, Avoiding, is responsible for collision detection and avoidance.

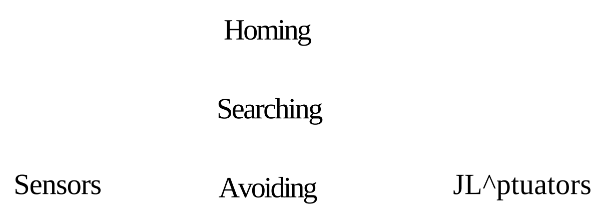


Figure 3.2 Behavior Based Control Structure

Obstacles in a foraging environment include the perimeter boundary and other robots. Collision detection with environment boundary is achieved via a touch sensor. In response to the activation of a touch sensor, robot moves directly backward for a short distance and rotates counter-clockwise through a random angle. While a robot detects another robot within touch sensor range, both robots stop first, backward then turn a random angle to resume it's previous behavior.

Searching fulfills the task of searching the environment for available food. A simple search method consists of following a straight path until a collision occurs. The

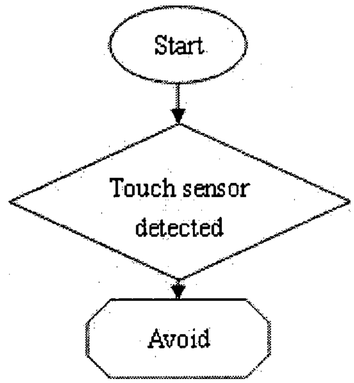


Figure 3.3 Level 1: Avoiding

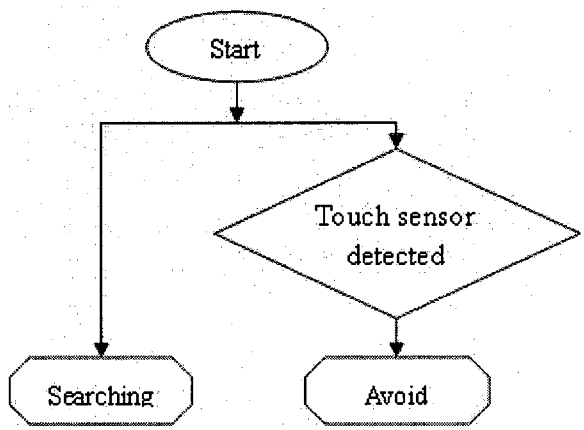


Figure 3.4 Level 2: Searching

avoiding behavior then produces a new initial heading, at which point straight searching motion resumes. The sensors (the touch sensor and IR sensor) are monitored during searching: touch sensor senses collisions, and the IR sensor looks for the food.

Homing takes place when a robot finds food or runs out of energy. Robot goes home by reading from a compass and setting a heading direction toward home. Once

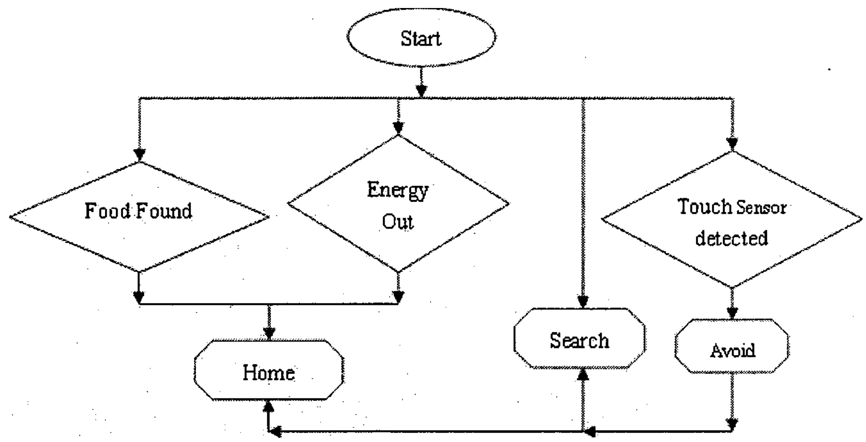


Figure 3.5 Level 3: Homing

gets home, robot releases food at home then rotates approximately 180 degree and ready to start searching behavior again.

As these basic behaviors complete control structure for a robot to produce the desired foraging behavior.

3.3 Collective Foraging Simulation Program

Computer simulation is a common method in the study of a robotic system. This is because computer simulations are easier to setup, less expensive, faster and more convenient to use. They tend to be far simpler to implement than experimenting with real robots, especially for a swarm robotic system which ideally have a large number of robots. Computer simulations can be use to validate the control algorithm at earlier age of the desing and predict the behavior of robot group which is still very difficult in physical experiments.

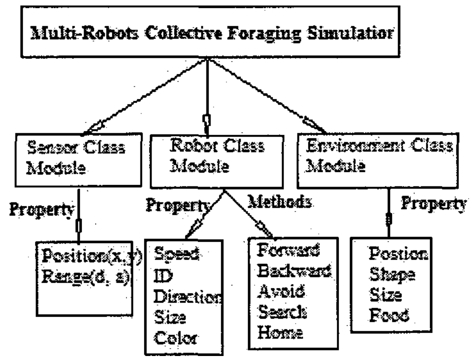


Figure 3.6 Multi Robots Foraging Simulation Program

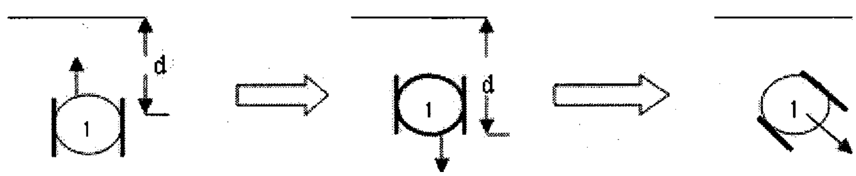
In this study of swarm robotics, simulation programs were developed and used to visualize the collective foraging behavior as well as to validate the proposed design. It is also used to enhance the understanding of the optimized group behavior where mathematical models have not yet developed. The detailed introduction is provided as follow.

We first developed a class module for each object needed in the simulation program. Each class module serves as a template for an object. For example, from a robot module, 12 robot objects are generated for collective foraging task. Class module has properties which are used to describes the attributes of the object and methods to carry out the actions on the object. With different property or method, objects from the same class module could behave differently with different methods.

Figure 3.6 lists all the class modules: robot, sensor, and environment and property and methods for each class module in simulation program. Robot module is shown with the ID number in the circle. The arrow shows the forward direction of a robot.

Class module of the environment is a square shape with foraging area on the top and home area at the bottom. We use a small circle with a color represent available food in the simulation. In the case an environment with more than one type of food, different type of food is represented by different sizes or color.

All major behavior of a robot was defined in methods of the robot module: avoiding, searching and homing. Figure 3.7 shows how a robot avoid environment boundary in the simulation. Touch sensor can detect boundary within sensor range *d*.



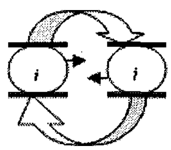
i) robot detect object   ii) robot backward one step   iii) robot turn random

Figure 3.7 Robot Avoids Collision with Boundary

Figure 3.8 shows a robot searching for a food with IR sensor in simulation program. Two attributes of sensor are distance D and angle a. If a food located within the sensor range, sensor will detect it.

Figure 3.9 shows how two robots encounter and interact with each other. When two robots are closer within a distance which is measured by a sensor, both robots first stop to avoid collision, and then both robots will backward for two steps and then set a random direction to resume their previous behavior.

Figure 3.8 Robot Detects Food within Sensor Range (D,a)



(1) exchange information (2) avoid each other (3) go random direction

Figure 3.9 Robot Interacts with Another Robot

Homing behavior is shown in Figure 3.10. A robot goes forward to the home direction which is provided by a compass.

^^Home direction

Figure 3.10 Robot Going Home

3.4 Finite State Machine of Foraging Activity

A finite state machine was used to show all the states of robots in foraging activity in Figure 3.11. Each square represents a state which robot is in during foraging process. Transitions between two states occur on the base of events that are either external (triggered by sensors) or internal (e.g. energy out) to a robot. The labels on the edges in the graph show the action or evens happen between states. The states and events are explained as follows:

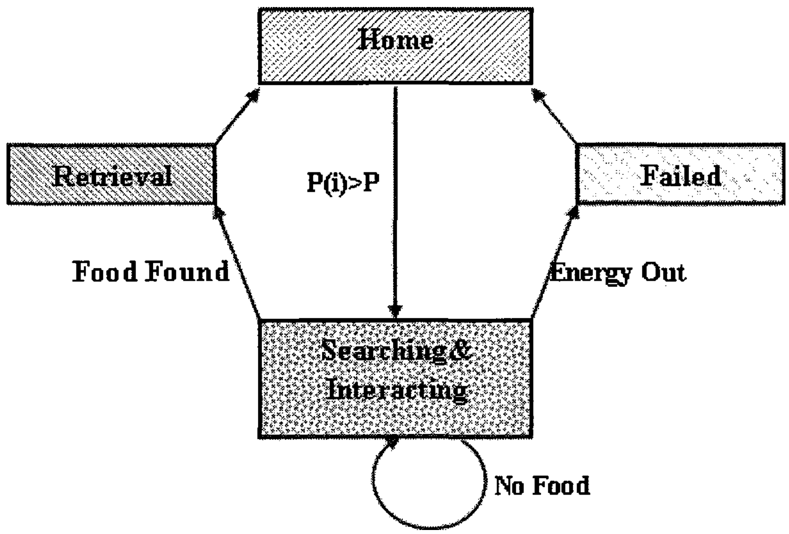


Figure 3.11 Finite State Machine of Robot Foraging Behavior

- **Foraging probability** $Pi>P_0$ (Event): If foraging probability of robot i higher than the predefined value Pq, robot i exits home and goes into searching state.

- **Searching and interacting** (State): A robot randomly searching in a foraging area. Meanwhile, when two robots are close within a sensor range, they stop and exchange state information, then avoid the collision. Robots then resume to previous foraging state.

- **Failed** (State): Once a robot is in failed state, it goes directly back home.

- **Retrieval** (State): Once a robot find a food, it goes back home with the food.

- **Energy out** (Event): If a robot costs more energy than predefined searching energy limit before finding a food, foraging task is failed.

- **Food Found** (Event): A robot finds a food within energy range.

All robots are in one of the states which were listed above.

In order to help users of the simulation program easily visualize foraging states of robots, foraging robots show different colors when they are in different foraging states. Figure 3.12 is a screen shot of a collective foraging simulation. In the foraging area, robots with red colors are in successful retrieval states, robots with black color are in actively searching states and robots with yellow colors are in failed states. It also shows several robots staying at home and rest robots activly foraging in the foraging area.

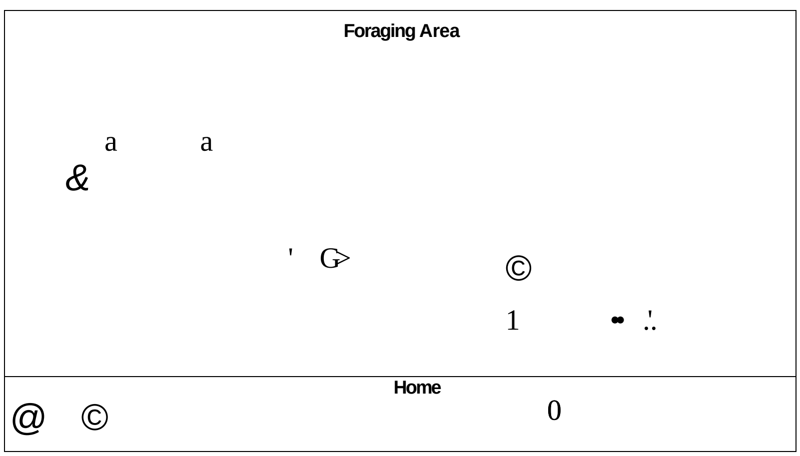


Figure 3.12 Screenshot of Collective Foraging Simulations

3.5 Mechanism of Division of Labor

3.5.1 Division of Labor in Social Insects

From Section 3.1 we know that in order to achieve group goal of maximizing net energy from collective foraging activity, the desired group behavior is to have an optimal number of robots active foraging and rest robots staying at home. In other word, optimal division of labor between foraging and resting among robots is the desired group behavior.

Self-organized behaviors, such as division of labor or specialization, in which individuals within a group perform different roles, can be observed widely in social insects group, along with the large number of ants follow some simple behavioral rules. They are known to divide their tasks amongst groups of workers (called castes) to improve task efficiency. In addition, one of the intriguing features of their behavior is that they are adaptive to environmental situations and able to react to changing demand by switching tasks to those for which the colony demand at that instant is high. It is also fundamental to their ecological success.

Take a group of ants called Pogonomyrmex barbatas as a good example of division of labor and task switching in social insects. These ants are known to have four different castes (patrollers, foragers, nest maintenance workers and madden workers), each having a particular task. Patrollers are one of the first groups of workers to emerge in the morning. They do most of the trail laying and also assess whether it is safe to forage or not. The successful returns of the patrollers trigger the foragers to emerge out of the nest. Foragers use the direction chosen by the patrollers and ignore food sources that were not explored by the patrollers. The nest maintenance workers are involved in building and maintaining nest chambers inside the nest while the madden workers accumulate the refuse pile or madden and move it from one place to another. Red harvester ants not only carry out their own tasks but also to change to a different task if required. This changing ability comes into action in response to a increased demand for a particular task. For instance, when there is a flood and the nest gets damaged, the need for more nest maintenance workers would become high. In such circumstances ants from inside the nest task switch and start working as nest maintenance workers. If there is a need for foragers, all the other castes can switch their tasks to foraging and assist the foragers.

A simple response threshold model has been used by several authors to explain the division of labor in social societies. ([105, 106, 107, 108, 109]) In this model, it is assumed that individuals in a group are characterized by response thresholds to various tasks stimulus. When the intensity of a stimulus associated with a task exceeds the response threshold of a worker, that worker engages in task with high probability when it perceives the stimulus. Task performance reduces the intensity of the task stimulus, thereby decreasing the probability that other workers engage in the same task. Response threshold can be adjusted by reinforcement. Mathematical model of response threshold is given in Eq.(3.2). X is the state of a individual. (X=0 correspond to inactivity, X=l correspond to the state of actively performing the task), 8i is an internal threshold of individual i response to task. S is the perceive task stimulus. An inactive individual i starts performing a task with probability of $P_{\%}$:

$$PI(X= 0 => X = 1) = \wedge T e f \quad (3\text{-}2)$$

Figure 3.13 shows the probability curves corresponding to the response threshold func-

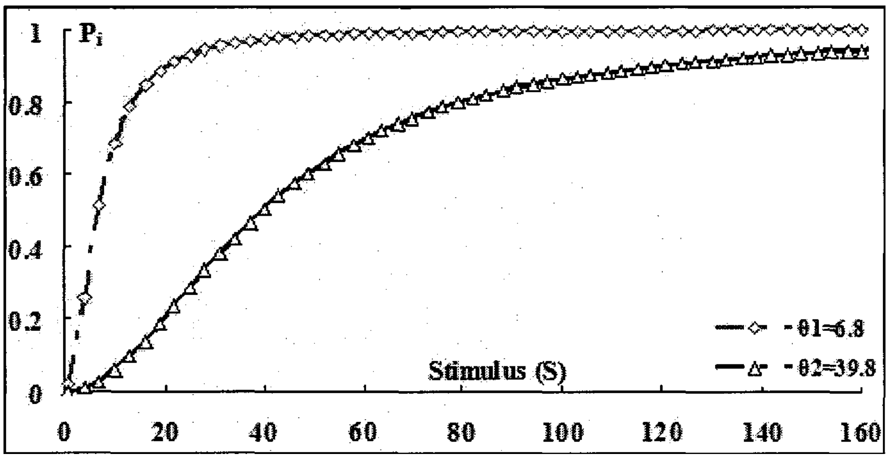


Figure 3.13 Probability Curves of Response Threshold Model

tion. As it is shown in the plot, P "the probability of individual performing the required task"goes higher with increasing stimulus S perceived from the task. The variations in intensity of stimulus can be caused by **(i)** task performance of other individuals, which decrease stimulus intensity; **(ii)** natural changes of task demands irrespective weather or not task is performed. Under the same intensity of S, individuals with lower response thresholds (0) have higher probabilities of performing the required task.

3.5.2 Division of Labor in Collective Foraging

In the study of collective foraging in a environment with one type of food, robot either actively forage to retrieve more food energy or rest at home to save energy of the group. The desired group behavior in this study is to have an optimal division of labor between active foraging and resting among robot so that group foraging performance can be improved.

In order to regulate the behavior of a robot between actively foraging and resting, the concept of probability of active foraging is adopted. A robot with a higher foraging probability forages more often than a robot with a lower foraging probability. Response threshold function is used to calculate the foraging probability of a robot /': *P(i)* shown in Equation(3.3),

$$p(,) = <FTMf \qquad (3.3)$$

Here *S* is the stimulus which is used to indicate the availability of the food in the foraging environment. *Th(i)* is the response threshold of robot *i* to the foraging task. With different foraging probabilities among robots in the group, division of labor among robots is emerged at group level which is shown Figure 3.14. When the foraging threshold of an individual robot or the stimulus of foraging task changes, foraging probability of the robot will be updated and the number of active forager in the group changes accordingly. As the result of the process, number of active forager at the group level can be adjusted in a self-organized way.

3.6 Collective Foraging Performance Metrics

In order to investigate weather or not foraging performance is improved, several foraging performance metrics has been defined. From Section 3.1, we know that in

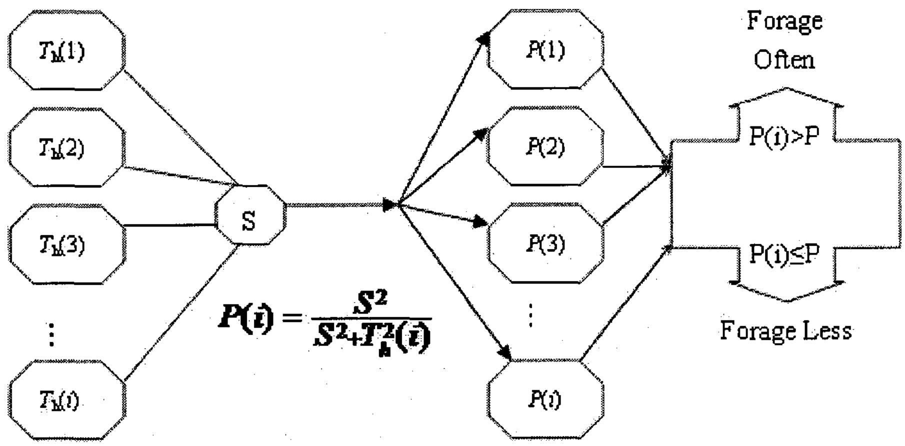


Figure 3.14 Mechanism of Division of Labor

collective foraging activity, from energy efficiency point of view, the goal of the group is to acquire as much net energy from environment as possible by having an optimal number of robot foraging. In previous studies, foraging performance was only measured by the energy efficiency of the group. In this study, group foraging performance includes energy, time, retrieval efficiency as well as group flexibility, and robustness while foraging in a dynamic environment. Follows are the explanation of each metric which is used in this study.

- **Energy Efficiency**

Energy efficiency is used to measure what percentage of available food energy in the foraging environment can be acquired by foraging robots as net energy. It considers both energy cost and income of the group. In the same environment, group with higher energy efficiency can acquire more net energy. Group energy efficiency in this study is defined as in Equation(3.4)

$$\text{ti} = \frac{FoodEnergy\ Retrieved - GroupEnergyCost}{AvailableEnergyInFoodSource} \tag{3.4}$$

Here Available energy is calculated using food energy growth rate (energy units/mins) times simulation time.

- **Time Efficiency**

Next metric we use to measure group foraging performance is time efficiency. The calculation of the time efficiency is defined in Equation (3.5). Group duty time is the sum of time which foraging robots spend on foraging activity. Time efficiency measures how fast the robot group retrieves net energy from food source in the environment.

$$\frac{FoodEnergy\ Retrieved - GroupEnergyCost}{GroupDutyTime}$$

- **Retrieval Efficiency**

Another quantitative measurement of group performance is retrieval efficiency. It measures how powerful a robot group retrieves food energy from food source in the environment without considering energy cost of the group. A robot group could has a high retrieval efficiency but low energy efficiency when the energy cost is high. This metrics is also used to find out weather design can help robot group retrieve more food energy from food source in foraging environment.

$$\frac{FoodEnergy\ Retrieved}{AvailableEnergyInFoodSource} \tag{3.6}$$

- **Group Flexibility**

To a swarm robotic system, group flexibility refers to the group's respond to internal perturbations or external challenges. In the context of collective foraging in this study, external challenge comes from a dynamic change of food source in the environment. We test the flexibility of the robot group in an environment with a dynamic changing food source.

- **Group Robustness**

Robustness of a swarm robotic system means that the desired group task can still be accomplished even when some robots in the group failed. Group demonstrates redundancy and high fault tolerance to partial failure. Experiments were set up to show the robustness of the design group in the event of some of foraging robots suddenly fail to perform foraging task.

- **Group Scalability**

We test scalability of the group with various group sizes on performing the collective foraging task. Weather performance of the group affected by size of the group measures the scalability of the group. Number of interactions between robots in the foraging area also directly relates to the group size with in turn affect the speed of adaptation.

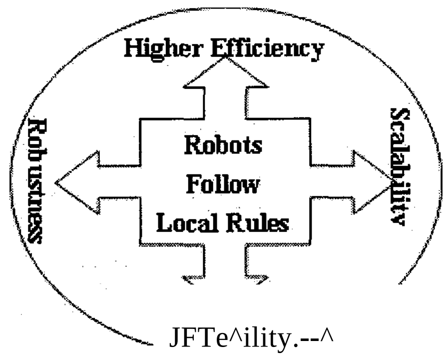


Figure 3.15 Desired Robots Collective Foraging Performances

3.7 Measurements

In each experiment, all the variables related to group foraging performance is sampled at the rate of 0.5Hz (1 data every 2 minutes) and recorded in a separate file. We run each experiment 10 times and use average value for data analysis.

The first step on analyzing experiments data is to find the steady state of foraging process. The number of active foraging robots in each foraging process indicates weather division of labor is emerged at group level as well as the steady state of this group behavior. In this study, best fitting curve is used to fit data from experiments to a predefined function curve. Once the best fitting function is found, the steady state value of experiment data could be estimated from steady value of the fitting function. Figure 3.16 shows an example of a curve fitting process. The average number of active foraging robots from the experiment is plotted with red dots. Dotted black line is the best fitting curve for this experiment data. The function of the fitting curve is y=10.81 exp(-0.035f)+3.1, and steady state value from this function is

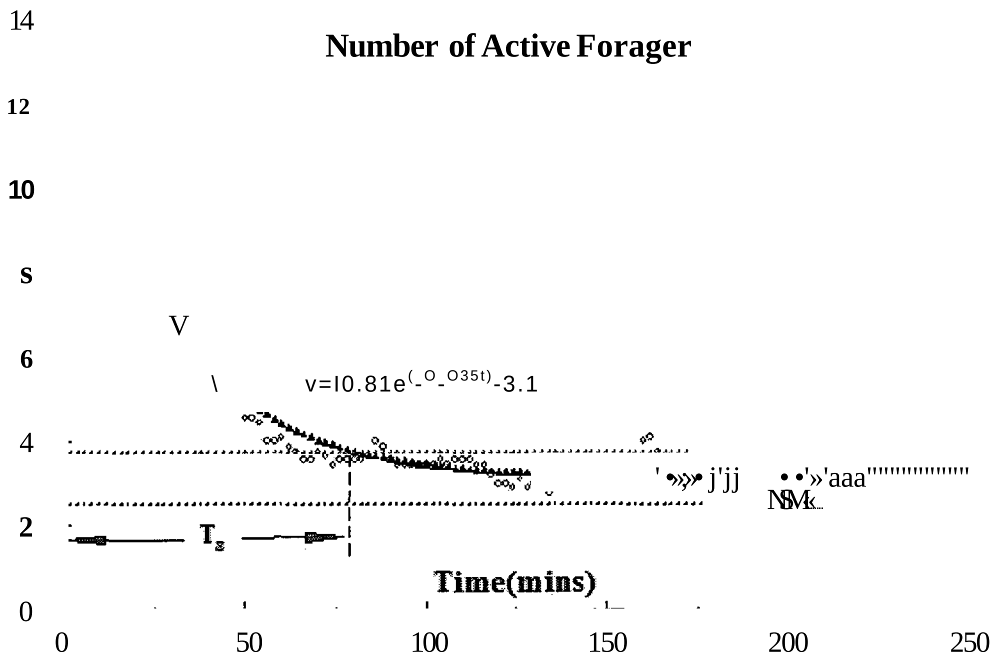


Figure 3.16 Experiment Data with Fitting Curve

$y(t \to \infty)=3.1$. This also considered as the steady state of number of forager in the foraging process. Once steady state value is determined, settling time *(T_s)* is a variable which is used to measure the time from start of a foraging process to number of active foraging robots settle within ±10% of the steady state value. Delay time is a variable which is used to measure how long it takes robot group response to a step changes of the food source in foraging environment.

CHAPTER IV

HOMOGENEOUS COLLECTIVE FORAGING

In Section 3.5, a foraging threshold model has been introduced to determine foraging probability for each robot. Robots with a higher foraging probability will active foraging more often. In the group, some robots forage more times and others forage less. Division of labor between foraging and resting is emerged at robot group level. Since food availability in the environment is unknown, each robot has to follow certain rule adapt values of foraging threshold and foraging stimulus so that number of active foraging robots can be optimized. In other word, an adaptation rules needed for each robots to adapt foraging probability so that optimal division of labor can be emerged at group level.

In this chapter, a set of adaptation rules has been designed for robots through the interactions between foraging robots and environment and between foraging robots. The adaptation rules are introduced in Section 4.1 in details. Section 4.2 explains numerical simulations using an excel model of the design. Several sets of experiments have been designed to validate the proposed design in Section 4.3. Results from the experiments are discussed in Section 4.4. Section 4.5 summaries the results and conclusions.

4.1 Interaction and Adaptation Rules

Recent research shows that a system with a self-organization usually relies on some basic ingredients: (1) positive feedback, (2) negative feedback, (3) multiple interactions. The positive feedback typically promotes the creation of structures or pattern and amplifies the fluctuations of the system. The negative feedback often serves as a regulatory to counterbalance positive feedback and stabilize the adaptation process. Both positive and negative feedbacks are the results of interactions between individual and environment or among individuals. Multiple interactions play an essential role for a desired collective behavior to emerge.

In a foraging activity, a robot interacts with the food source in environment every time it forages for food. A successful food retrieval can be consider as a positive feedback and a failed foraging trip is a negative feedback from the interactions. While this robot in the foraging area, every time it encounters with another robot can be considered as an interaction opportunity.

The adaptation rules for interactions between foraging robots in foraging area are designed as follows:

- **Interaction Rules:** take robot i as an example: while robot i moving in the foraging field, if it encounters with another robot, here take robot j as an example, they will exchange foraging state information with each other and record it in a variable called taskcounter(«). As robot i keep interacting with other foraging robots, taskcounter(0 accumulates the foraging information of other robots it has encountered in the environment. At beginning of each foraging trip, value of taskcounter is reset.

Table 4.1 Interaction Rules in Foraging Environment

Robot / encounters Robot *j*
If robot *j* is at **Retrieval** state Then taskcounter(0=taskcounter(/);
If robot *j* is at **Searching** state Then taskcounter(/)=taskcounter(/)-l;
If robot *j* is at **Failed** state Then taskcounter(/)=taskcounter(0-2;

- **Adaptation Rule:** When robot *i* gets home. The net energy from the foraging trip can be determined. Only when the net energy of the trip is positive, it called a successful foraging. Robot *i* updates foraging threshold Th(i) and foraging task stimulus *S* according to it's own foraging performance and information recoded in taskcounter(i).

The above rules implement the reinforcement mechanism for the foraging probability of the individual robot. Robot *i* adapts *Th(i)* according to the results from the interactions between itself and food source in the environment. At the same time, it adapts *S* according to the results from the interactions between itself with other foraging robots as well as between itself with food source which is recorded in taskaccount. With adapted Th(i) and *S*, foraging probability of robot *i*, *(P(i))* will be adapted too. In a foraging environment which has only one type of food available, results from both interactions indicates food source situation, e.g. availability of food.

The selection of values for adjustment and attenuation factors A_{15} A_2, \$1, \$2 and in the algorithm will depend on the environment setting and the parameters of the

Table 4.2 Adaptation Rules at Home

Adaptation Rule for T_h of Robot *i* and *S*	
Foraging threshold of robot *i*: $T_h(i)$	If robot *i* success Then T/l(?')=T/1(z)-A1 If robot *i* failed Then $T_h(i)=T_h(i)+A_2$
Foraging stimulus for the group: *S*	If robot *i* **Success** and taskcounter(/)=0 Then: S=S+$i If robot *i* **Not Success** and taskcounter(/)<0 Then: *S=S-§*2 If robot *i* **Failed** and taskcounter(z')<0 Then: *S=S-$*3

robots behaviors in simulations. With careful selection of those factors each robot can adapt its foraging probability resulting in task switching between foraging and resting. The desired division of labor emerged and collective foraging performance is improved.

4.2 Spreadsheet Model

With the mechanism of division of labor and adaptation rules have been designed for individual robot, we are ready to run the simulations to verify the design. First we defined all the variables. Table 4.3 lists all the variables and symbols which are used in the simulations. For each robot in the group, there are some variables needed to be calculated and recoded in the foraging process. Table 4.4 lists all the variables and calculations related to each individual robot, here take robot *i* as an example. All

Table 4.3 Symbols Were Used in Computer Simulation

Name	Symbol
Robot Group Size	m
Robot ID	i
Foraging Trip ID	j
Number of Trips of Robot i	k(0
Energy Acquired per Food	u
Food Growth Rate	G_new
Moving Energy Cost	e
Energy Cost per Interference	c
Searching Cost Limit	I
Foraging Stimulus	S

calculations of group variables related to the collective foraging process is listed in Table 4.5. These variables are used to calculate group foraging performance.

Group foraging performance metrics which have defined in (3.1)-(3.3) now can be updated with symbols of variables by equation (4.1)-(4.3).

$$\wedge \quad \frac{FoodEnergy\,Retrieved - GroupEnergyCost}{AvailableEnergylnFoodSource} \qquad \frac{ul_n - eT - cE_n}{G_{new}ut} \tag{4.1}$$

$$\frac{FoodEnergy\,Retrieved - GroupEnergyCost}{GroupDutyTime} \qquad \frac{ul_n}{T} \qquad \wedge \quad \wedge$$

Table 4.4 Variables of Individual Robots in Simulation Program

Variables	Symbols
Moment to Start Foraging of Robot *i* on j^{th} trip	*toutihj)*
Moment of Backing Home of Robot *i* on j^{th} trip	*tback(i,j)*
Number of Interferences of Robot *i* at j^{th} trip	$E_n(i, \quad j)$
Total Interferences of Robot *i*	fc(i) $^{E}n(i) = Y.^{E}n(hj)$ j=1
Duty Time of Robot *i* on j^{th} trip	$T(i,j) = t_{in}(i,j) - t_{back}(i,j)$
Food Collected of Robot *i* at j^{th} trip	*In(i,j)*
Foraging Threshold of Robot *i*	$T_h(i)$
Foraging Probability of Robot *i*	*P(i)*

Table 4.5 Calculations of Group Variables in Simulations

Name	Group Variables
Number of Active Forager	*n*
Group Duty Time	*m* *T=J2T(i)* i=i
Total Food Collected by Group	*m* *In = Yl ^(i)* i=l
Total Interference	*m* $E_n = J2^{E}n(l)$ t=l

FoodEnergy Retrieved I_n

AvailableEnergylnFoodSource $G_{new}t$

Once all the variables and calculations were defined, we did numerical simulations using mathematic functions in the spreadsheets of Excel. Each robot in the group has a table like Table 4.6 which has all the variables and calculations for this robot in foraging simulation.Once we plug in all the initial values and adaptation factor, all local variables of this robot can be automatically updated by Excel. There are total 12 robots in the group, so there are 12 tables like Table 4.6 for the group. Finally, Table 4.7 is used

Table 4.6 Calculation Table for Individual Robot; Ai=A_2=0.**1**; 3>i=3>$_2$=0.2

j	*tcmt(i,j)*	*^backij"!*	*j In(i,j)*	*En(i,j)*	*T(i,j)*	*Th(i,j)*	*P(i,j)*
1	*2*	26	1	1	24	20.2	0.502
2	28	54	0	3	26	20	0.5
3	55	61	1	4	6	20.2	0.5
k(i)							
					i s j a T & j)		
			In(i)	$E_n(i)$	*T(i)*		

for calculating all the group variables. All local variables of each individual robot in this table are from each robot table. With the help of numerical simulations described in this section, all the calculations for foraging process are verified. We are ready to validate design in computer simulation programs.

Table 4.7 Calculation Table for Robot Group; Ai=A_2=0.1, $\$i=\$>_2=0.02$, $\$_2=0.01$

Robot ID	*Inii)*	$E_n(i)$	T(i)
1	11	5	94
2	13	9	116
m			E£iT(z)
	In	E_n	*T*

4.3 Experimental Set-up

In computer simulation, at the beginning of each experiment, all robots are at home with initial value of foraging threshold: *Th{i)*=19.5+Rnd (Rnd is a random number generated by computer, 0<Rnd<l) and initial foraging stimulus (5=20). In order to maintain the food density in simulation environment at a reasonably constant level over the experiment time, new foods are placed randomly in the searching arena with G_{new} the growth rate, food units per minute. By changing the G_{new}, we can obtain a desired food source density in foraging environment. The number of available food energy can also be calculated through G_{new} and experiment time. Once a food is found by a robot, it will be removed form environment to prevent other robots from retrieving the same food. In each time step of the simulation, a robot will consume an amount of energy while it moving. Table 4.8 lists the energy consumption setting for the different behavior in the simulations. The sum of energy consumed on moving and interferences

Table 4.8 Energy Units Consumption of Different Activities

Activity	**Energy Units**
Moving	5/step
Interference	16/step
Retrieval	4000/food
Searching Limits	3800/trip

with other foraging robots in a foraging trip is the energy cost for a robot. Taking into account the energy cost in a foraging trip for a robot arid the energy income from a successful retrieval (4000 energy units), in order to make a positive contribution to the group net energy, the maximum energy cost of a robot on one searching activity alone has to be less than 4000 units. We set the energy limit for searching activity alone S_e is 3800 units considering energy cost on returning home. This means that if a robot already consumed 3800 units of energy on searching alone and still could not find a food. Robot is considered in failed state and goes back home. S_e is set to prevent the robot from wandering outside home for too long, which could result in a huge energy cost to the group energy even if the robot does eventually retrieve a food.

There are no obvious guidelines for selection of adaptation factors: A_{15} A_2, <&i, $\$_2$ and $3 in the algorithm. We therefore choose these values on a trial and error basis. Generally, a large change in foraging threshold *(Th(i))* and foraging stimulus *(S)* will potentially cause oscillation or stabilization problems for the group while a small change could lead to a slower adaptation process. Table 4.9 summarizes all of the setting for the adaptation factors were used in the experiments. In order to investigate

Table 4.9 Selection of Adaptation Factors

AI	A_2		\$2	$\$_3$
0.02	0.05	0.001	0.005	0.002

whether, and how, proposed adaptation rule could help robot group reach a desired optimal division of labor, two types of experiments are designed and tested first, which first vary the population of robot in the group and then the food source density in the environment. Control group, in which no design are taken into account provides us a benchmark group with fixed number of foraging robots.

4.4 Experiment Results

4.6.1 Robot Groups With Various Populations Collective Forage in a Given Environment

The first set of experiments were designed to simulate several robot groups with various populations collective forage in a given food source with a fixed G_{new}=2. Robot groups with 3, 6, 9 and 12 robots forage in the same enviornment setting. For each group, we run simulations using both control and design group 10 times, and each experiment last for 250 minutes. The simulation program records the number of food retrieved, net energy of group accumulated, group duty time and number of active foragers during the simulation, and we then average data of 10 runs' for analysis.

(i) Steady state of division of labor

We first need to find out the steady state of division of labor among robots in each foraging process. This is determined from the instantaneous number of active foragers in each experiment. In the robot group with 3 and 6 robots, data shows that design groups always have all the robots actively foraging because there are enough food for all robots foraging. The steady states of division of labors in both two groups are having all robots actively foraging. Instantaneous number of active foragers in both experiments is plot in Figure 4.1. However, in the groups with 9 and 12 robots, there

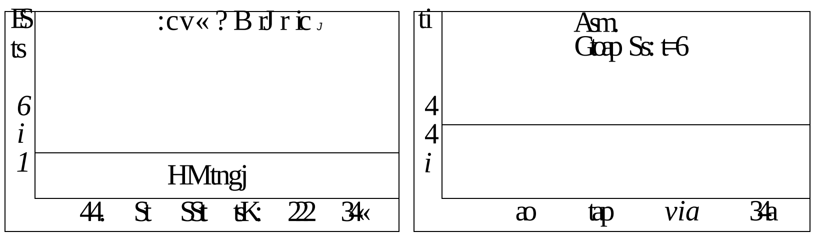


Figure 4.1 Number of Active Forager in Design Groups with 3 and 6 Robots

is not enough food for all robots foraging all the time. Division of active foraging and resting among robots emerged at group level. Figure 4.2 left shows the instantaneous number of active foragers of experiments in the group with 9 robots. Curve fitting process has been used to determine the steady state value of number of active forager. From best curve fitting function, the steady state value of active forager in this process is 6.1. The number of foragers oscillates within ±10% of steady state value after 135 minutes which indicate foraging process is in steady state. The design group has about 6 robots foraging and rest robots resting at home to save energy. The same procedure was used to analyze the design group with 12 robots. Instantaneous number of active

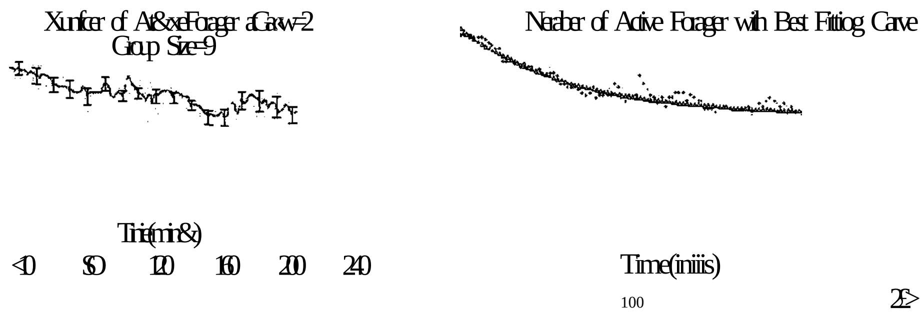


Figure 4.2 Number of Active Forager in Design Group with 9 Robots

foragers in this foraging process is plot in Figure 4.3. The steady state value from fitting function is 5.98 and settling time is 78 minutes. From this set of experiments, it shows

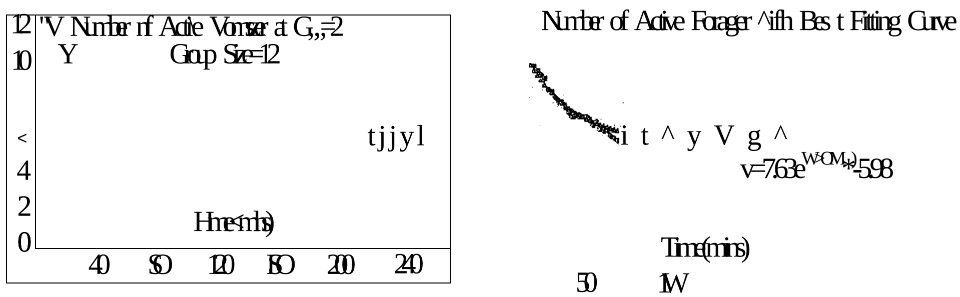


Figure 4.3 Number of Active Forager in Design Group with 12 Robots

that in the design group, division of labor between active foraging and resting among robots emerged at steady state of each collective foraging process. The plot in Figure 4.4 summaries the instantaneous number of active foragers of all groups in this given environment *(*G_{new}=2). From the settling time of groups with various populations, it indicates that the adaptation process is faster in the group with a larger group size. But the steady state of division of labor among robots is mostly depends on the food source in environment. In this case, since the G_{new} is fixed, there will be always about division

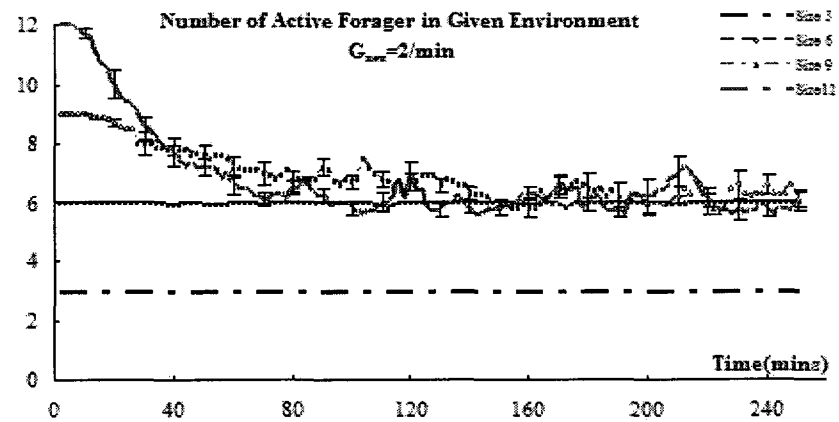


Figure 4.4 Instantaneous Active Foragers of Design Groups with Various Populations

of 6 robots active foraging and rest robots staying at steady state no matter what group size is.

ii) Optimal group foraging performance

Division of labor which is emerged at group level has been confirmed in last section. Here we investigate weather the divisions of labor optimized group foraging performance compare to the control group. In other word, we need to find out weather group has a desired division of labor. We first plot the instantaneous energy efficiency and time efficiency of the both design and control group from each foraging process in Figure 4.5. It shows that in the group with fewer robots (3 or 6), since all robots active foraging all the time, there is not big gap between design and control groups. Especially in the group with only 3 robots, there is almost no difference on energy and time efficiency between design group and control. As group size grows, design groups show highly improved energy efficiency and time efficiency compare to control group. All the foraging efficiency matrices of both design group and control group at steady state

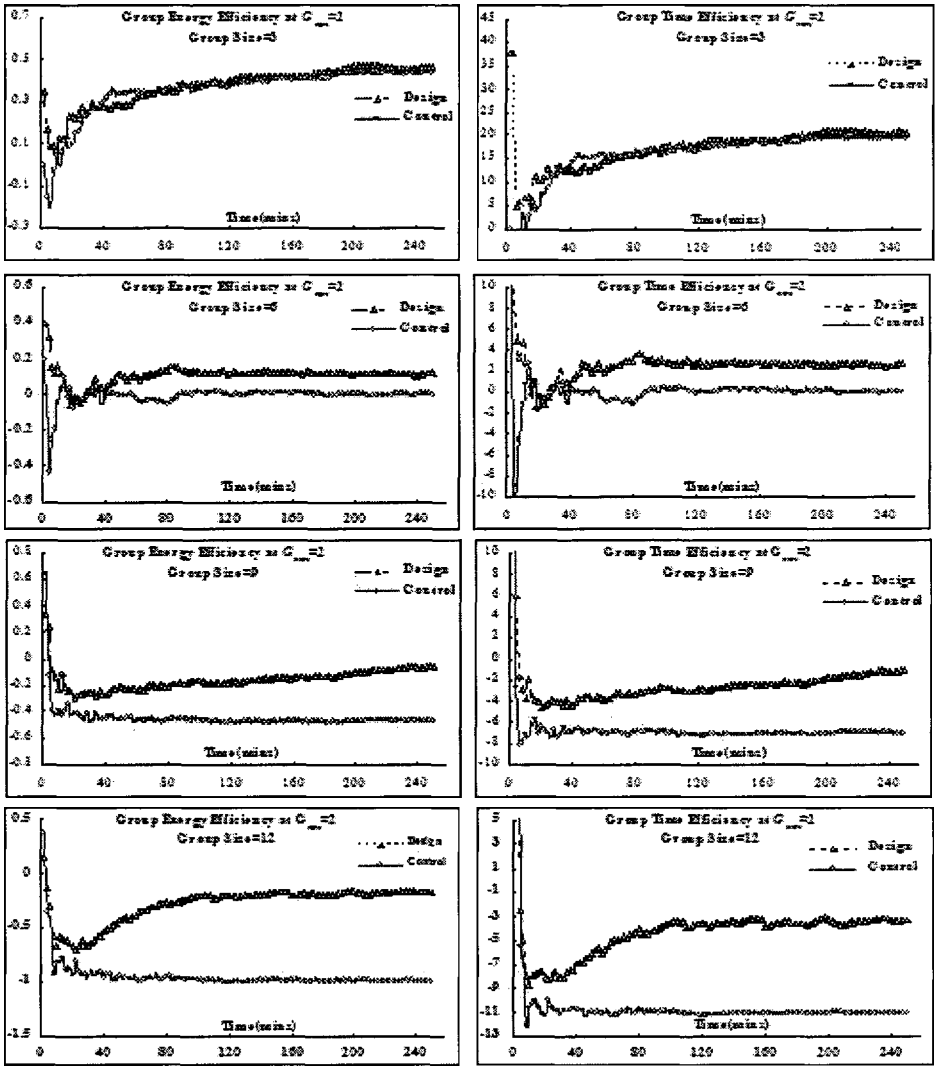


Figure 4.5 Instantaneous Time and Energy Efficiency of Design and Control Groups with Various Populations of Robots Forage in a Given Environment

are calculated and list in Table 4.10. Each Simulation Lasts for 250 Minutes. With the optimal division of labor, design group has improved foraging performance in energy and time efficiency compare to control group. However, there is no big gap in term of retrieval efficiency between design and control group. It shows design group is same powerful to acquire energy from food source as control group even with less forag-

Table 4.10 Average Results from 10 Runs of Robot Groups with Different Populations Foraging in a Given Environment. The Food Density Remains the Same During Each Simulation (G_{new}=2).

Group Size	Strategies	Energy Efficiency %	Time Efficiency %	Retrieval Efficiency	Active Forager
3	Control	43.7	19.52	93	3
	Design	46.4	20.75	88.4	3
6	Control	2.6	0.06	98.4	6
	Design	11.7	2.63	97.8	6
9	Control	-47	-7.01	99	9
	Design	-7.8	-1.43	98.4	6.2
12	Control	-99	-11.03	99.8	12
	Design	-17.2	-3.4	97.6	6.1

ing robots. It also tells us that design group achieve higher energy and time efficiency because of save energy cost instead of acquire more energy.

4.6.2 A Robot Group Collective Forage in Differeent Environments

We designed a second set of experiments to investigate the effect of adaptation rules when design group forage in different environment conditions; we fix the population of group to 12 robots but run the experiments with three different food source densities, from poor (G_{new}=1) to relatively richer environment (G_{new}=4). Again each experiment is run 10 times and each simulation lasts for 250 minutes.

(i): Steady state of division of labor in different environments

We first find the steady state of each collective foraging process in these experiments. The instantaneous number of active foragers in each foraging process and best fitting curve are plot in Figure 4.6. In the food source with G_{new}=1, the steady value

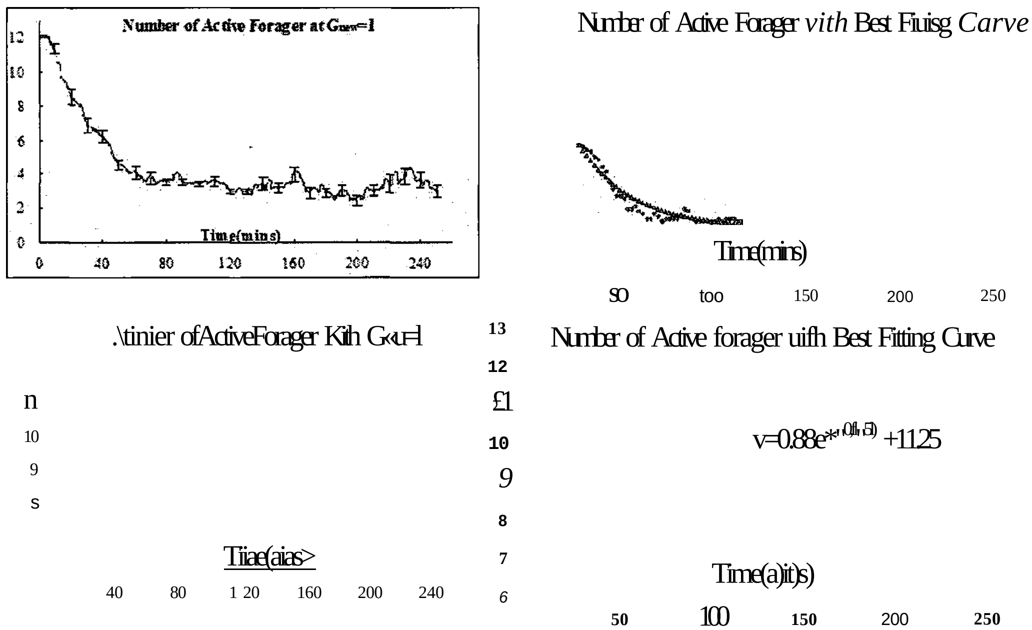


Figure 4.6 Instantaneous Active Foragers in Design Group with 12 Robots at Environment with G_{new}=1 and G_{new}=4

of active forager of design group from fitting function is 3.1. The similar process determines the steady value of active forager in food source with G_{new}= 4 is 11.25. We summarize and plot the instantaneous number of active forager of this group at different environments in Figure 4.7. With all the robots (12 robots) active foraging at the beginning, groups have different number of active foragers at steady state in different environments. In a relatively richer environment (G_{new}=4), there are around 11 robots actively foraging at steady state. At a poor environment (G_{new}=1), there are

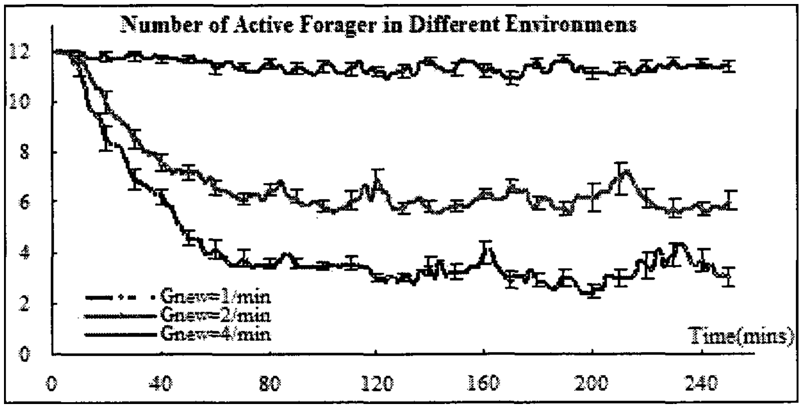


Figure 4.7 Instantaneous Active Foragers of Design Group with 12 Robots in Different Environments

around 3 robots actively foraging and rest robots staying home at steady state. It shows the flexibility of the group by having different numbers of active foragers in different environments.

(ii): Optimal group foraging performance

Instantaneous value of energy and time efficiency of design and control group in different environments are plot in Figure 4.8. As we expected, design group shows a much higher group efficiency compare to control group at all foraging environment settings. It also shows that the gap between design and control group becomes smaller with a higher G_{new} which means a higher food density in foraging environment. The reason is in a environment with higher food source density, a robot is more likely to find food and hence reward itself (with a higher foraging probability) to forage more often. In a food source with G_{new}=4, most robots in design group will actively forage at steady state since there are enough food in the environment. The results from calculations of foraging performance metrics under different environments are provided in Table

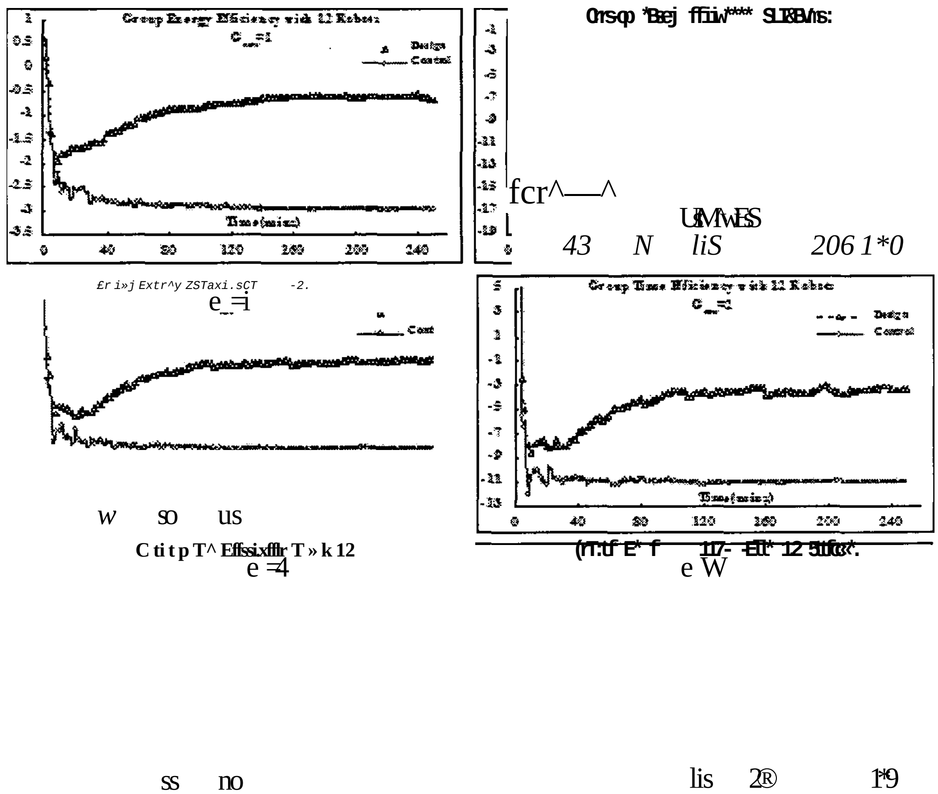


Figure 4.8 Instantaneous Time and Energy Efficiency of Design and Control Groups with 12 Robots Forage in Different Envrionments

4.11. All the results are calculated using steady state value. Each Simulation Lasts for 250 Minutes. The design groups have highly improved foraging efficiency both in time efficiency and energy efficiency. There is no big gap between design group and control group on retrieval efficiency. Both groups are same powerful to retrieve most food from food source in environment. It also indicates us the reason design group has an optimized foraging performance mainly because of design group has less foraging robots in an environment with a poor food source.

$$AverageRetrievalTime = \frac{GroupDutyTime}{FoodRetrievedByGroup} \qquad (4.4)$$

Table 4.11 Average Results from 10 Runs of a Robot Group with 12 Robots Foraging in Different Environments.

Growth Rate(G'neiu)	Strategies	Energy Efficiency (%)	Time Efficiency (%)	Retrieval Efficiency	Active Forager	Retrieval Time (minutes)
1	Control	-297	-16.61	99	12	712
	Design	-62	-9.15	94.5	3.2	286
2	Control	-99	-11.03	99.8	12	358
	Design	-17.2	-3.4	97.6	6.1	206
4	Control	-4.3	-0.96	98.6	12	181
	Design	14.6	3.342	97.6	11.3	176

We plot the average retrieval time of design group under different environment, from poor to rich, in Figure 4.9. Compare to the control group, average retrieval time of design group is quite stable which implies robot group with the adaptation mechanism is quite robust to environmental changes.

4.6.3 Collective Foraging in a Dynamic Changing Environment

To test the flexibility of the design group in a dynamic changing environment, we step up a dynamic changing food source in the environment. A robot group with population of 12 robots initially engaged foraging in a environment with G_{new}=1, then we disturb the environment by introducing a step change of G_{new} from 1 to 4 at t=217 minutes. Then again G_{new} changes from 4 to 2 at 433 minutes. Each experiment we

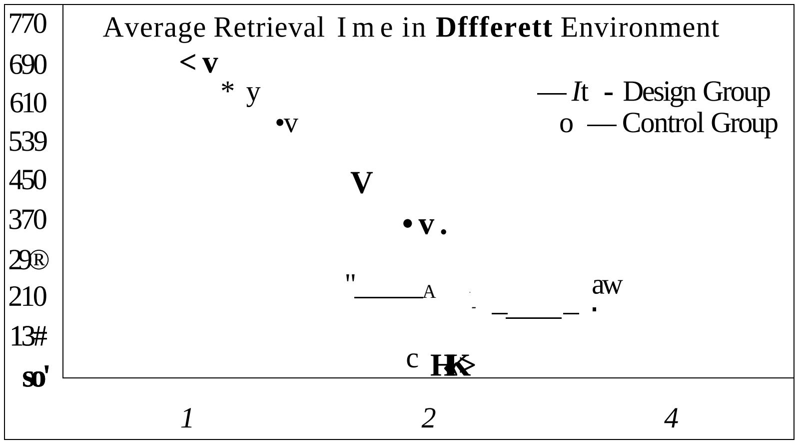


Figure 4.9 Summary of Average Retrieval Time of Design and Control Groups with 12 Robots Forage in Different Environments

repeat 10 times and each simulation lasts for 750 minutes, with all parameters of the group remaining the same.

We plot the instantaneous number of active foragers in Figure 4.10. As expected, a new dynamic equilibrium for the number of active foragers in the design group is observed each time after food source changes. Design Group has more resting robots to save energy when food source becomes poor and more foraging robots when food source is rich. It shows not only the division of labor among robots always can emerged in a dynamic changing environment but also group's flexibility to response to an external dynamic changes in the foraging environment. Each time food source changes, after some delay, an optimal division of labor will emerge at steady state in design group. During the simulation, design group acquires more net energy from food source. Figure 4.11 compares the instantaneous net energy accumulated in design group to control group. In an environment with less food energy available (G_{new}=1), design strategy

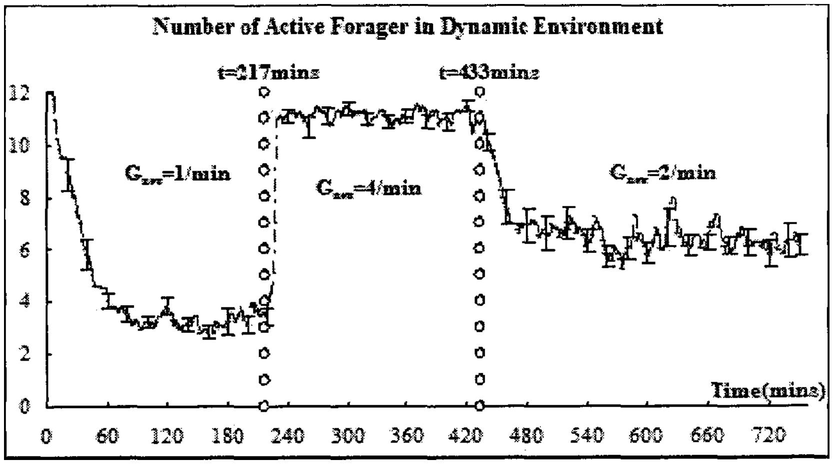


Figure 4.10 Instantaneous Number of Foraging Robots of Design Group in a Dynamic Changing Environment

can keep group from losing too much energy by having less foraging robots. Once the food source in environment become rich, group collectively perceive the changes and sending more foraging robots to acquire more energy from food source.

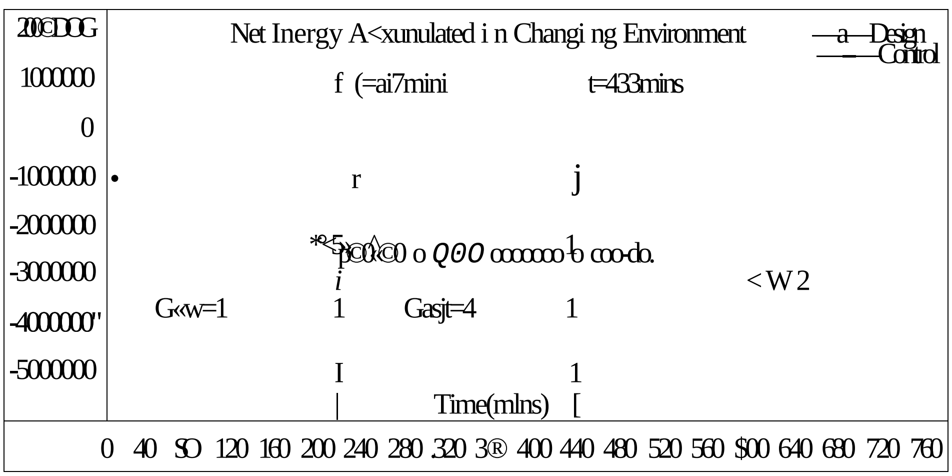


Figure 4.11 Instantaneous Net Energy Design Group Acquired From a Dynamic Changing Environment

The next set of experiments is designed to test if the design group shows fault tolerant. Unexpected foraging function disturbance (sudden step change of foraging threshold of some foraging robots) happens at t=100 minutes. Only few robots survived and keep foraging. Simulation lasts 250 minutes. We plot the instantaneous number of active foragers in foraging arena in Figure 4.12. From the data, it shows that even a few forging robots survived, after some delay, some robots that previous resting at home start foraging again. There is more food accumulated in the environment because of insufficient number of foragers. The number of active foraging robots in the environment reaches a new steady state eventually. The data shows that design group with the adaptation mechanism is robust to its internal function disturbances.

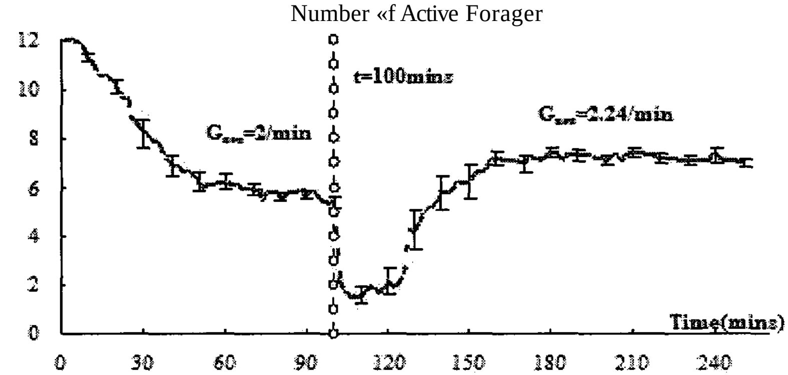


Figure 4.12 Number of Active Forager in Design Group When Some Foraging Robots Disfunction at t=100 Minutes

4.6.5 Different Adaptation Rules

In Section 4.3, we designed a set of adaptation rules for each robot to adapt foraging probability through the interactions between foraging robot and environment and between foraging robots. As the result, desired collective behavior of division of labor is emerged and group foraging efficiency is optimized. Robot group also shows flexibility to the dynamic changes of food availability in the environment and robustness to the function failure of some foraging robots.

In the experiments, there is only one type of food available in the foraging enviroment, the interactions between robot and environment provide direct cue about food availability through robot own foraging experience. The interactions between foraging robots also provide cue about food availability through other foraging robots experience. Proposed adaptation rules use both cues to adapt the foraging probability of individual robot. In order to determine how each cue alone affects the division of labor of in the group, two strategies - each with only one cue - are designed for each robot to adapt. Strategy S_r use only internal cue which is the result from interactions between robot and environment to adapt foraging probability of robot, and strategy S_e use only social cue with is results from interactions between foraging robots to adapt foraging probability of robot. The adaptation rules of each strategy S_r and S_e are defined are as follows in Table 4.12: We designed a set of experiments to investigate weather the adaptation rules by strategy S_r and S_e can also help robot group reach a desired division of labor at group level. The experiments is set up to run in a given environment with G_{new}=1 first. Two robot groups with different group sizes collective foraging in the given environment using different adapt strategies: S_r and S_e. Figure 4.13 plots the number of active forager in the group using different adaptation strategies. Strategy S is the proposed adaptation strategy which used both cues to adapt. Other than number

Table 4.12 Different Adaption Strategies for Individual Robot

Adaptation Rule for $T_h(i)$ and 5		
Strategy	S_e (internal cue only)	^(social cue only)
	If robot *i* success Then $T_h(i)=T_h(i)$-$A_{i;}$ 5=5+$!	If taskcounter(i)=0 Then $T_h(i)=T_h(i)$-A_2; 5=5+3>$_2$
	If robot *i* failed Then $T_h(i)=T_h\{i)$+$A_{i;}$ 5=5-$x	If taskcounter(i)<0 Then $T_h(i)=T_h(i)$-A_2; 5=5+<&$_2$

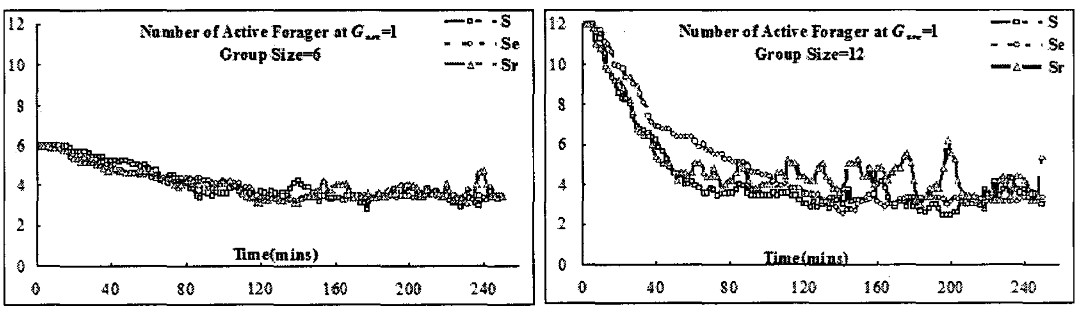


Figure 4.13 Instantaneous Number of Active Forager of Robot Group Using Different Adaption Strategies Forage in a Given Environment

of active foragers in each group using different adaptation strategy, settling time and steady state error are used to compare results from experiment. In a robot group with 6 robots, there is no big gap among different adaptation strategies in term of settling time and steady state value of active forager. It shows in a group with less population can achieve desired division of labor at steady state either from social cue or internal cue. As group size grows, the number of interactions between foraging robots grows. In a robot group with 12 robots, groups use different adaptation strategies have gaps on settling time and steady state error. Table 4.13 lists the settling time and standard deviation of error calculated from the robot group with 12 robots using different strategies. Group using strategy S_r has shortest settling time but highest error at steady state.

Table 4.13 Steady State Error and Settling Time of the Same Robot Group Using Different Adaptation Strategies

Adaptation Strategy	**STD (Steady State)**	**Settling Time (minutes)**
S	0.4	64
Se	0.24	120
S<r	0.66	50

On the other side, Group using strategy S_e has longest settling time but lowest error at steady state.

We test a robot group with 12 robots using different adaptation strategy in a dynamic environment with step changes of food source density. Results from the experiments shows in Fig 4.14.

Delay time is used to measure how long it takes robot group to adapt to the environment changes. Delay is start from t=217 minutes when step changes of food source happens first. Table 4.14 lists the delay time of the group using different strategies. It shows that adaptation through only inter cue is slower but more stable and the adaptation through only social cue is faster but less stable. The effect from social cue related to group size because more foraging robots, more interactions between foraging robots.

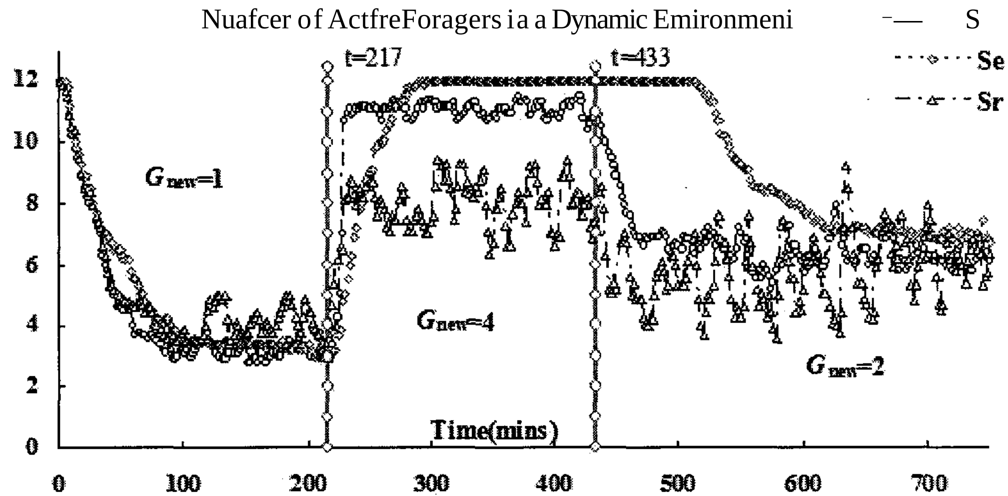


Figure 4.14 Instantaneous Number of Active Forager of Robot Group Using Different Adaption Strategies Forage in a Dynamic Environment

Table 4.14 Delay Time for Robot Groups with 12 Robots Using Different Adaptation Strategies

Adaptation Strategy	Delay Time (minutes)
S	21
S_e	79
S-p	23

The proposed adaptation rules use both cues to help robot group reach the desired division of labor in faster but also with less steady state error.

4.5 Summary and Conclusions

In this chapter, we have proposed a simple adaptation mechanism for a group of robots in performing a collective foraging task. Individual robot follows a set of designed adaptation rule while foraging for food in a bounded area, collective behavior of division of labor is emerged. As the results, highly improved collective foraging performance is achieved at group level.

The individual robot in the group use internal cues (from interactions between foraging robot and environment), social cues (interactions between foraging robots) to determine whether they will rest in the nest for longer (lower foraging probability) to either save energy or minimize interference, or be actively engaged in foraging more often (with higher foraging probability) which potentially gains more food energy back to the group. With the designed mechanism, the group demonstrates:

- Improved foraging performance (time efficiency and energy efficiency) compared to the control group.
- Robustness to failure of individual foraging robots.
- Flexibility to adapt to dynamic environment changes.
- Scalability to keep group performance with different group sizes.
- Emergent desired group behavior of optimal division between active foraging and resting among robots in the group.

Furthermore, the group with the adaptation mechanism is able to guide the system towards optimization of energy efficiency despite the limited sensing abilities of the individual robots and the simple interaction rules. The group also exhibits the capacity to perceive the environment collectively if we consider the average number of active foragers in the environment over time. That is, more foraging robots indicate a richer food source and less foraging robots indicate a poor food source. This correlation can only be observed at the overall group level and cannot be deduced from individual robots.

Some other interesting conclusions are, firstly, the division of labor at steady state from foraging process is directly related to the food source availability in the environment. In a given environment, robot groups with different populaions using adaptation rules will have same division of labor at steady state.

Secondly, that since individual robot only communicates and adapts locally, the same algorithm can be use in robot groups with different sizes. The design group has scalability.

Finally, social cue from multiple interactions between foraging robots help design group speed up the adaptation process. The bigger robot group size, the more foraging robots which result in more interactions between foraging robots, the faster adaptation process is.

CHAPTER V

HETEOGENEOUS COLLECTIVE FORAGING

In chapter 4, we have validated the adaptation rules designed for a group of simple robots collective forage for one type of food in a environment. Without a center control, desired group behavior-division of labor between foraging and resting among robots is emerged at group level in a self-organized way. As result of this, group foraging performance is highly optimized. In this chapter, we have extended collective foraging task to where there is more than one type of food available at food source in foraging environment. Individual robot is able to forage for any type of food follows the same states as describe in Figure 3.11. but a robot only can forage one type of food at a time. This means foraging robots behave differently in the environment according to the type of food and robot group becomes a heterogeneous group.

The goal of the collective foraging activity keeps the same as in homogeneous case: acquire net food energy from food source in foraging environment as fast as possible. But the desired group behavior changed since now there is more than one type of food available in the foraging environment. The desired group behavior in this system becomes to have an optimal task allocation of foraging robot on different types of food as well as an optimal division of labor between foraging and resting among robots. As results of this desired group behavior, collective foraging performance expected to be highly optimized. We have extended the adaptation rule that individual robot in the group now is able to switches its behavior not only between foraging and resting but also between two types of food.

This chapter is organized as follows: the detail of desired group behavior in a heterogeneous collective foraging problem is presented in Section 5.1. The extended adaptation rules have designed for robots are present in section 5.2. A set of experiments using computer simulation have been designed, experiments setup is introduced in section 5.3. Results from the experiments and analysis form the data is shown in 5.4. Summary and conclusions of the design are in section 5.5.

5.1 Task Allocation

When there is more than one type of food available in the environment, collective foraging performances of a group of robots not only depends on the number of active foragers but also on the allocation of foraging robots on different type of food. For example, if there is a rich source of food A and poor source of food B available in the environment. The desired group behavior in this environment is a task allocation of more foraging robots on food A and less foraging robots on food B. With desired task allocation as well as division of labor, group can collectively achieve improved foraging performance.

In order to have both division of labor and task allocation emerged at the group level, we first upgraded the mechanism of division of labor from last chapter. Figure 5.1 illustrates the mechanism to achieve the task allocation and division of labor at robot group level with help of foraging response model. Each robot now has two foraging

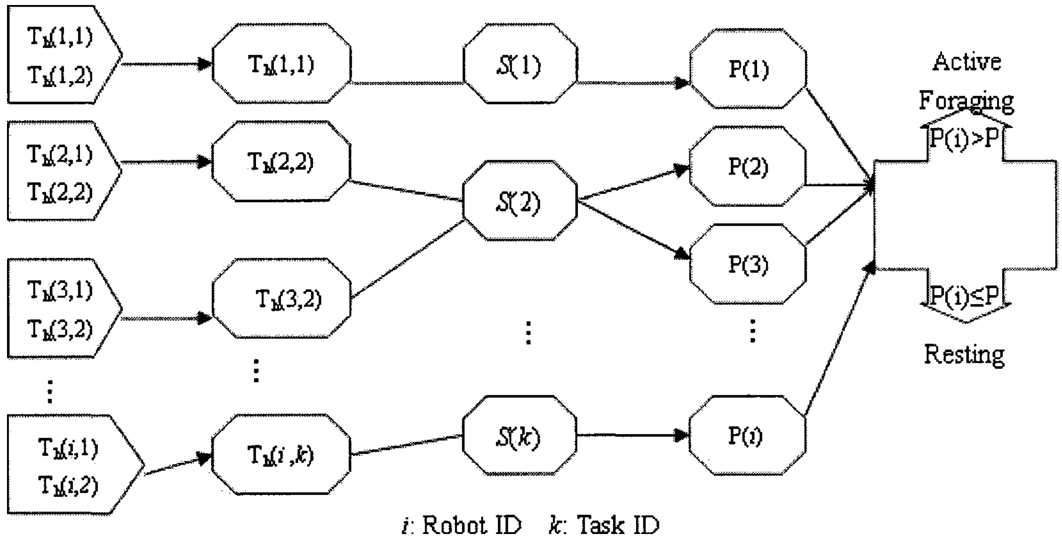


Figure 5.1 Mechanism of Task Allocation and Division of Labor

thresholds, one for each type of food: *Th(i, k)* (£=1, 2) here / is the robot ID and *k* is food type ID. There are also two foraging stimulus (S(k)) which are used to indicate the availability of each type of food in the food source. Individual robot chooses lowest foraging threshold to update its current probability of foraging. Weather active foraging or resting at home depends on the foraging probability of the individual robot. For example, if *Th{i, k)* is the lower foraging threshold between two types of food available in the environment, then foraging probability of robot *i,* P(/),will be calculated by $T_h(i, k)$ and *S(k)* through threshold model. Robot *i* will foraging more often if *P(i)* is higher. Through the difference betwee foraging thresholds in a robot and difference on foraging probabilities among robots, task allocation on different types and division of labor between active foraging and resting can be emerged at the group level.

5.2 Adaptation Rules

Through the process shown in Figure 5.1, task allocation as well as division of labor can be emerged at group level. A set of adaptation rules were designed to adapt the foraging threshold of individual robot and the stimulus for two tpyes of food through the interactions between foraging robots and environment and between foraging robots. The proposed adaptation rules are explained as follows:

- **Interaction Rules:** while robot *i* encounter with robot *j* in the foraging area, they interact through exchanging foraging states, the adaptation rules are different depends on weather they are working on the same type of food or not.The interactions continue while robot *i* foraging in the area, taskcounter(f) records the accumulated foraging state information of other foraging robots. At the beginning of each foraging trip, value of taskCounter is reset.

- **Adaptation Rule:** When robot *i* gets home. The net energy from foraging trip can be calculated. Only if the actually net energy from foraging trip is above zero, it consider as a successful retrieval. Robot *i* updates foraging threshold $T_h(i, k)$ and foraging task stimulus $S(k)$ according to it's own foraging performance and value in taskcounter.

Each robot in the group updates $Th(i, k)$ and $S(k)$ according to results from multiple interactions between foraging robot and type *k* food and between foraging robots in the environment.

Table 5.1 Interaction Rules in Foraging Area

If robot *i* foraging on the **same** type food as robot *j*	If robot *i* foraging on **different** type food as robot *j*
If robot *j* is at **Retrieval** state Then taskcounter(0=taskcounter(0; If robot *j* is at **Searching** state Then taskcounter(0=taskcounter (/)-1; If robot *j* is at **Failed** state Then taskcounter(7)=taskcounter(/)-2;	If robot *j* is at **Failed** State Then taskcounter(i)=taskcounter(i)+1; Else taskcounter(/)=taskcounter(/)

Table 5.2 Adaptation Rules

Adaptation Rule for *Th(i, k)* and *S(k)*	
Foraging threshold: $T_h(i, k)$	If robot *i* **Success** Then $T_h(i, k)=T_h(i, k)-A_1$ If robot *i* **Failed** Then $T_h(i, k)=T_h(i, k)+A_2$
Foraging stimulus: *S(k)*	If robot *i* **Success** and taskcounter(i)=0 Then: S=S+$i If robot *i* **Failed** and taskcounter(/)>0 Then: 5=S-$\$_2$ If robot *i* **Failed** and taskcounter(0<0 Then: $S=S-\$_3$

5.3 Experimental Set-up

Figure 5.2 is a screen shot of computer simulation of a group robots collective foraging in a environment with two types of food. Foods are distinguished by their colors in the simulation program: red and blue. Foraging robot also display color according the type of food it foraging at the time. Foraging robots with different colors help user to visualize the task allocation among foraging robots at group level during the experiments. In order to simplify the problem, most definitions and symbols of variables

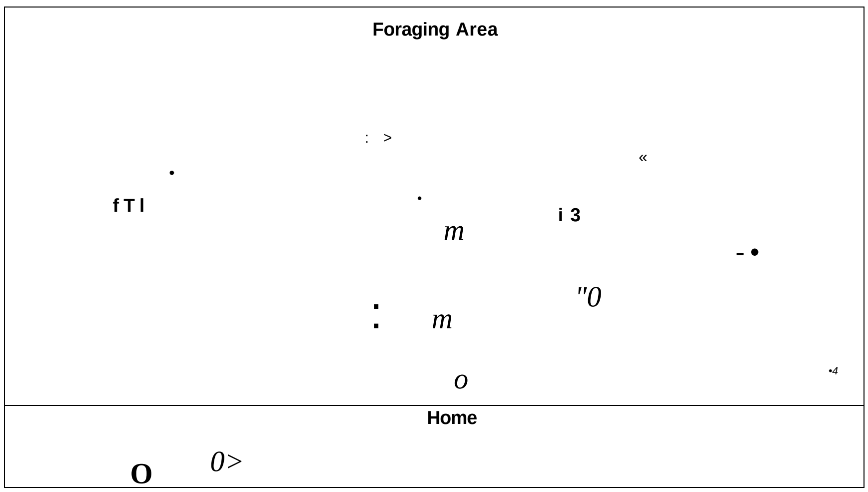


Figure 5.2 Screenshot of Heterogeneous Robots Foraging

listed in Table 4.1 remains the same. The variables related to food changes because there are two types of food in the simulations. We list the variables which have been changed in Table 5.3. Table 5.4 lists the settings for all the adaptation factors used in the adaptation rules. We choose these values on a trial and error basis. Generally, a large change in foraging threshold *($T_h(i, k)$)* and foraging stimulus *($S(k)$)* will potentially

Table 5.3 Variables Changes

Name	Homogeneous	Heteogeneous
Food Type ID	-	*k*
Foraging Threshold	$T_h(i)$	$T_h(i, \ k)$
Stimulus	*S*	*S(k)*
Food Growth	$G_n ew$	$G_n ew\{k)$
Number of Active Forager	*n*	*n(k)*

cause oscillation or stabilization problems for the group while a small change could lead to a slower adaptation process.

Table 5.4 Selection of Adaptation Factors

Po	AI	A_2	$>I	$2	$3
0.5	0.1	0.05	0.01	0.02	0.02

5.4 Experiment Results

5.4.1 Robot Groups with Variable Group Sizes Collective Foraging in a Given Environment

The first set of experiments are design to test group behavior when robot groups collective forage in a given environment with a fixed food density of each type: $G_{new}(red)=3$, $G_{new}\{blue)=1$ and $G_{new}=G_{new}(red)+G_{new}(blue)=4$. Here $G_{new}(red)$ is growth rate of red food and $G_{new}(blue)$ is growth rate of blue food. Groups with 8, 10 and 12 robots collective forage using designed adaptation rules in the same environment setting. For each group size, we run simulations using both control and design group 10 times, and each experiment last for 250 minutes. The simulation program records the number of food retrieved, net energy of group accumulated, group duty time and number of active foragers during the simulation. In the data analysis, we first determine the steady state of the each foraging process to see if task allocation or division of labor has emerged. Once steady state is confirmed, we average the 10 runs' data and calculate the group performance metrics for each experiment.

(i): Steady state of collective foraging process

The average instantaneous number of active foragers for each type of food of 10 runs and best fitting curves are plot in Figure 5.3. From best fitting functions, the steady state values of foraging robots on each type of food are determined. Since there are more red food available than blue food in the food source, with the adaptation rule, robots foraging for red food more likely to reward themselves and foraging more often on red food. In this environment, the total food growth rate is G_{new}=4. There is enough food for all the robots in the groups active foraging.

We run the robot group with 12 robots in a different environment with $G_{new}(red)$=1.8 and $G_{new}\{blue)$=0.6. In this environment $G_{new}=G_{new}(red)+G_{new}(blue)$=2.4. The ratio of $G_{new}(red)$ to $G_{new}(blue)$ is the same as previous experiment, but G_{new} is smaller so there is less food available. In Figure 5.4, we plot and compare the instantaneous

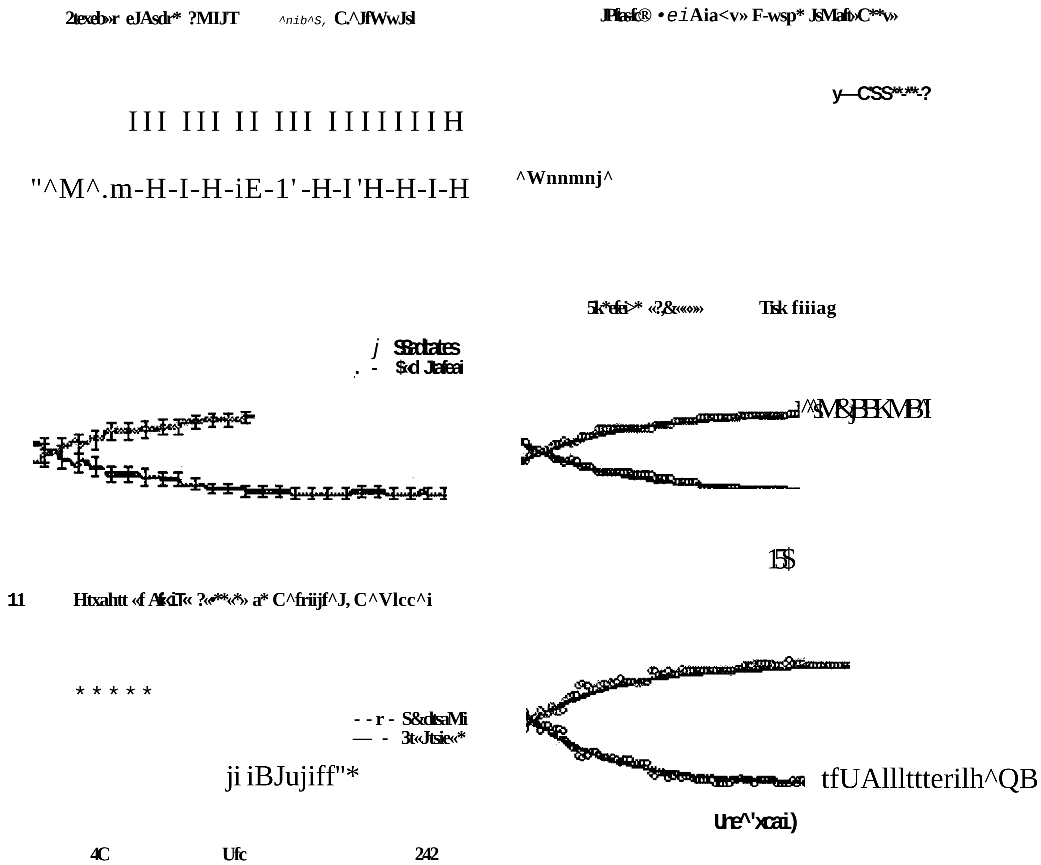


Figure 5.3 Instantaneous Numbers of Active Forager in Robot Group Population **8,** 10 and 12

number of active forager of each type as well as total number of forager of same robot group collective foraging in different environments. It shows this robot group has a

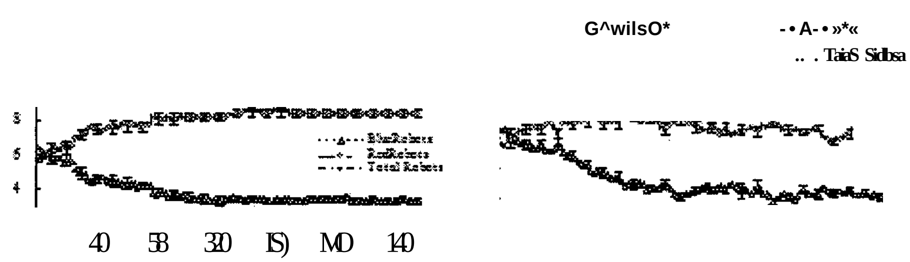


Figure 5.4 Instantaneous Numbers of Active Foragers of Design Groups with 12 Robots in Different Environments

same task allocation among foraging robots at the steady state which is more foraging robots search for red food. However, the division of labor at the steady state is different. In environment with $G_{new}=4$ all robots active foraging. In environment with $G_{new}=2.4$, there are average 8 foraging robots and rest robots staying at home. From this experiment, it shows that both division of labor and task allocation can emerge at group level when individual robot follows designed adaptation rules,

(ii): Optimal collective foraging performance

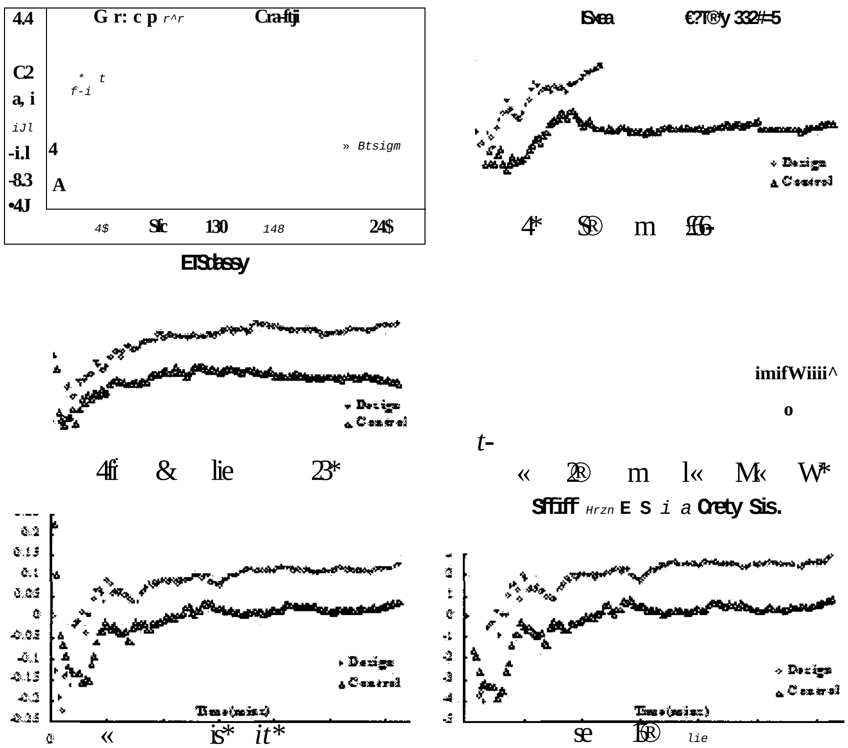


Figure 5.5 Instantaneous Energy and Time Efficiency of Design and Control Group Forage in a Given Foraging Environment: $G_{new}(red)=3$, $G_{new}[blue)=1$

With the help of multiple interactions between foraging robots and between robots and food source in the environment, task allocation of foraging robots on different type of food and division of labor between foraging and resting among robots emerge at group level. Figure 5.5 plots the instantaneous energy efficiency and time efficiency of both design and control group. After initial transient state, design group always can achieve a higher energy efficiency and higher time efficiency compare to the control group. Each experiment lasts for 250 minutes.

Table 5.5 lists the results from calculation of performance metrics for both design and control group. All performance metrics include energy efficiency, time efficiency and retrieval efficiency are all been significantly improved compare to results from control group. This shows the collective behavior emerged in design group is an optimal group behavior. Retrieval efficiency of design group shows that it acquires more food energy from food source than control group. The improved net energy of design group not only comes from division of labor but also from task allocation between two types of food in foraging robots.

5.4.2 A Robot Group Collective Foraging in Different Environments

We designed a second set of experiments to investigate the effect of designed adaptation rules when the robot group foraging in different environment conditions; here we fix the population of group to 12 robots but run the experiments in three different food sources: **(i)** $G_{new}(red)=3$, $G_{new}(blue)=1$; **(ii)** $G_{new}(red)=2$, $G_{new}\{blue)=2\backslash$ and **(iii)** $G_{new}(red)=3$, $G_{new}(blue)=1$. Total food growth rate in food sources are the same: $G_{new}=4$, but each food source has different ratio of red food to blue food. Each experiment runs 10 times and each simulation lasts for 250 minutes.

Table 5.5 Average Results from **10** Runs of the Heterogeneous Robot Groups with Different Populations. The Food Density Remains the same During Each Simulation(G_{new}(red)=3, G_{new}(blue)=1)

Group Size	Strategies	Energy Efficiency (%)	Time Efficiency	Retrieval Efficiency (%)	Active Red Robots	Active Blue Robots
8	Control	10	3.5	70.8	2.27	5.73
	Design	35	11.64	92.3	5	3
10	Control	11	2.93	81.2	2.34	7.65
	Design	30	9.73	90.6	6.66	3.24
12	Control	2	0.45	89.7	3.3	8.7
	Design	11	2.55	97.5	8.39	3.3

(i): Steady state of collective foraging process

We plot the average value of instantaneous number of active foragers with the best fitting curve in Figure 5.6. With the fitting curve function, steady state of foraging process can be determined in each experiment with different environment setting. It shows task allocation emerge in design group. In the environment where G_{new}(red)=G_{new}(blue)= 2, there are same amount of red food as blue food, active foragers split on two type food. But in environment setting where G_{new}(red)=1 and G_{new}(blue)=3, the task allocation in design group has about 8.1 foraging robots on the blue food and 3.2 on the red food.

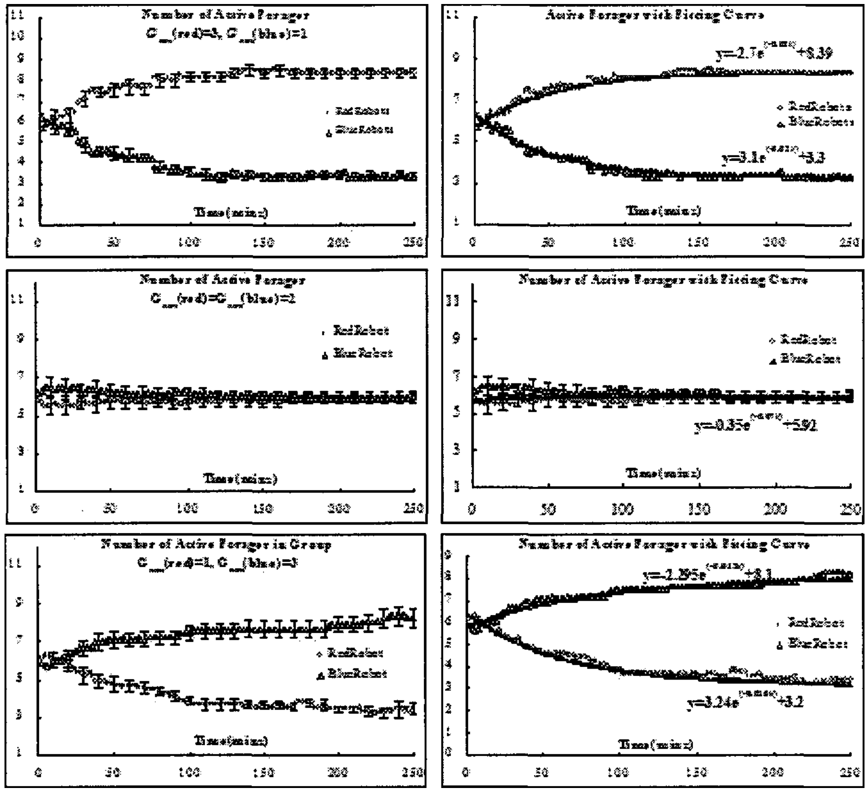


Figure 5.6 Instantaneous Number of Active Forager in Different Environments

Even individual robot does know the availability of food in foraging environment, with multiple interactions between foraging robots and environment as well as interactions between foraging robots, at group level, design group shows flexibility to adapt when environments changes.

(ii) Group foraging performance at steady state

We plot the instantaneous energy efficiency and time efficiency of both design and control group in Figure 5.7. In the environment which has *G_{new}(red)-G_{new}(blue)=2*, design group has same energy and time efficiency as the control group. This is becuase task allocation in design group is same as in control group. In other environment set-

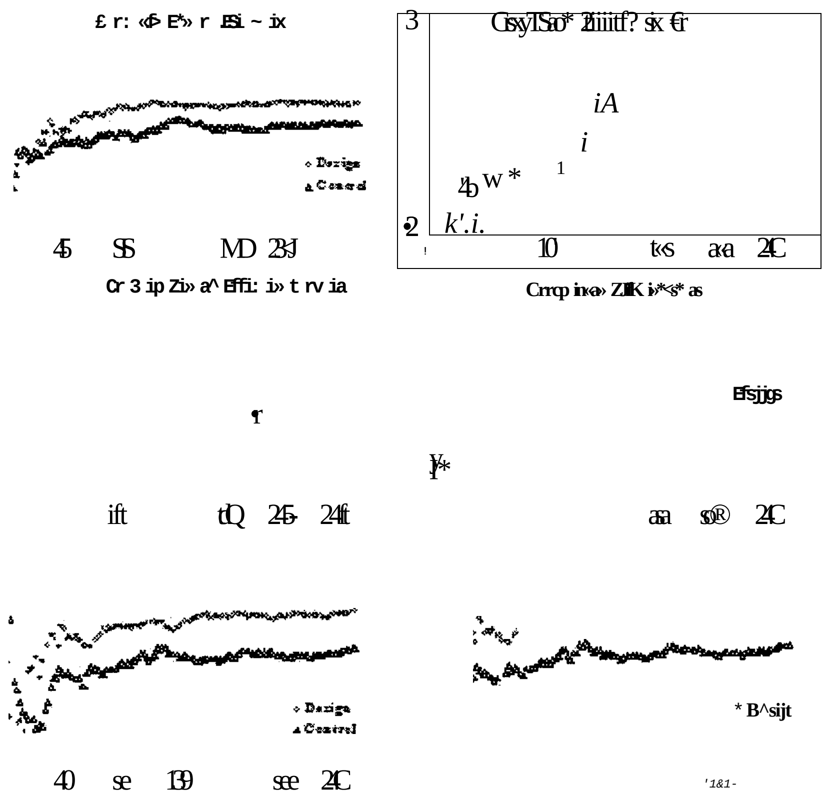


Figure 5.7 Instantaneous Energy and Time Efficiency of Design and Control Group Forage in Different Foraging Environments

tings, after initial transient state, task alloction at group level alway help design group achieve a higher foraging efficiency compare to control group. Table 5.6 summaries the results from calculation of performance metrics for both design and control group from average value of 10 runs. Figure 5.8 plots the average energy efficiency of both design and control groups in environments with different ratio of red food growth rate to blue food growth rate. It shows that in different environment settings, the average group efficiency for the design group is quite stable compared to control group, which implies the design group with the adaptation mechanism is more robust to the environmental changes.

Table 5.6 Average Results from 10 Runs of a Heterogeneous Robot Group with 12 Robots Collective Foraging in Different Environments

Food Growth Rate	Strategies	Energy Efficiency (%)	Time Efficiency	Retrieval Efficiency (%)	Active Red Robots	Active Blue Robots
$G_{red}=3$ $G_{blue}=1$	Control	2	0.45	89.7	3.3	8.7
	Design	11	2.55	97.5	8.39	3.2
$G_{red}=2$ $G_{blue}=2$	Control	12	2.62	98.2	6.31	5.7
	Design	12	2.59	98.3	6	6
$G_{red}=1$ $G_{blue}=3$	Control	3	0.68	90.1	8.15	3.85
	Design	11	2.46	98	2.62	9.38

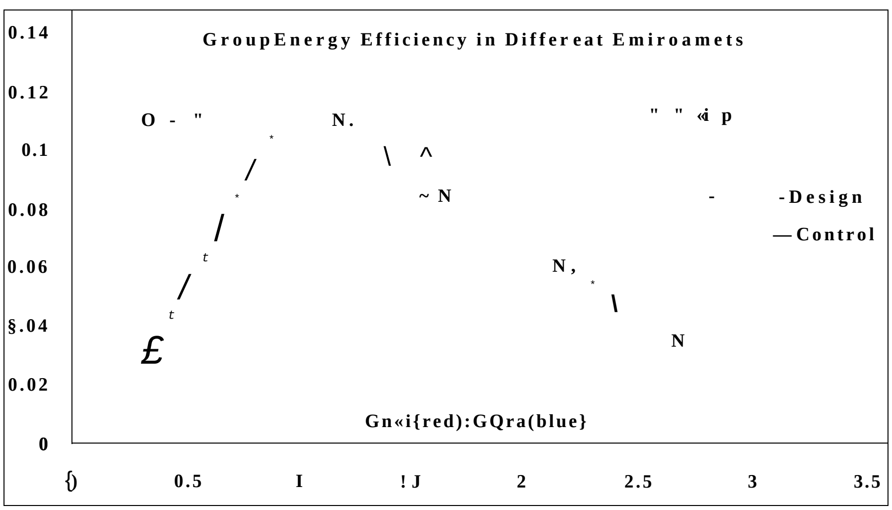


Figure 5.8 Average Energy Efficiency of Design and Control Group Forage in Different Environments

To test if the design group can adapt in a dynamic environment, we now disturb the environment by introducing a step change of food growth rate from $G_{new}(red)$=3.25, $G_{new}(blue)=0.15$ to $G_{new}(red)=0.15$, $G_{new}(blue)=3.25$ at t=250 minutes. A robot group with 12 robots, using design strategy, is engaged in a collective foraging task. Each experiment we repeat 10 times and each simulation lasts for 725 minutes, with other parameters remaining the same as above. We plot the average instantaneous number of active foragers for each type of food in Figure 5.9. As expected, a dynamic

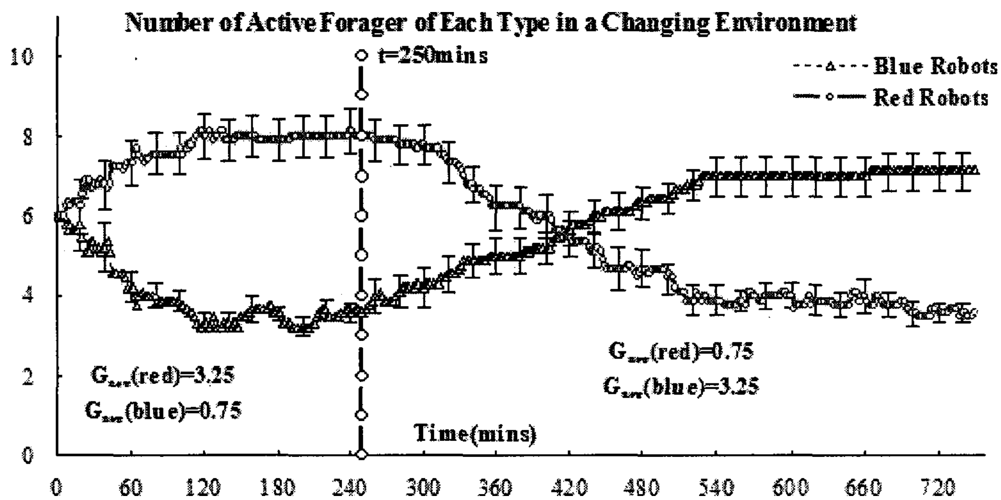


Figure 5.9 The Instantaneous Active Foragers of Design Group with **12** Robots Forage in a Dynamic Changing Environment

change of task allocation among foraging robots in the design group is observed. First, a stabilized task allocation emerged in design group with more red foraging robots and less blue foraging robots. When foraging environment changes at t=250 minutes, more foraging robots start to search for blue food and less foraging robots search for red food. After some delay, group reaches to a steady state with a desired task allocation among foraging robots. It shows robot group using the adaptation mechanism is able

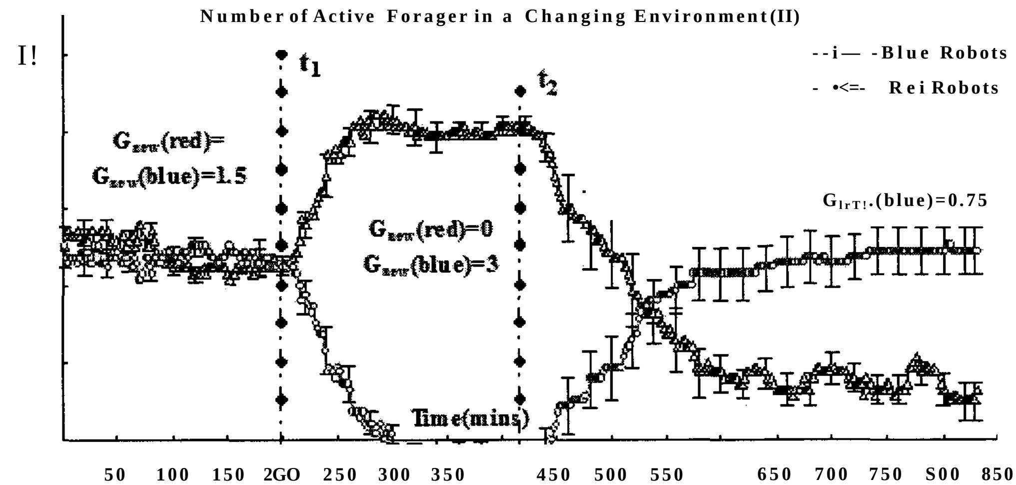


Figure 5.10 The Instantaneous Active Foragers of Design Group with 12 Robots Forage in a Dynamic Changing Environment Using Different Strategies

to collectively perceive dynamic changes of food density in the environment and adapt to the changes.

We designed another experiment with a dynamic environment change. We now disturb the environment by introducing a step change of food growth rate from initial setting of $G_{new}(red)$=$G_{new}(blue)$=1.5 to $G_{new}(red)=0$ and $G_{new}(blue)$=3 at t=200 minutes. It means start from t=200 minutes, environment only grows blue food and no red food any more. Food growth rate changes again to $G_{new}(red)$= 2.25 and $G_{new}(blue)=0.15$ at t=425 minutes again. The robot group has 12 robots, using use both design and control strategy, is engaged in the collective foraging task in this changing dynamic environment. Each experiment we repeat 10 times and each simulation lasts for 832 minutes, with other parameters remaining same as above. We plot the instantaneous number of active foragers on different type food with time in Figure 5.10. Initially, there are same amount of red food as blue food available in the environment. There is split number of foraging robots forage on two types of food. After environment changes at t=200 minutes, growth rate of red food changes to zero. There is less and less red food available. Group perceived this changes though the interactions and has less robots foraging for red food. After some delay, there are no robots foraging for red food any more, all foraging robots forage for blue food. When environment changes again at t=417 minutes, red food start to become available again. Design group perceived the environment changes again, after some delay; some foraging robots start to forager for red food again. Eventually, design group reaches a steady state with the desired task allocation.

5.5 Summary and Conclusions

In this chapter, we have extended collective foraging task from Chapter 4. A group of robots collective forage in a food source with two types of food. Foraging robot can only forage one type of food at one time. The goal of robot group is to acquire net energy from food source as fast as possible. The desired collective behavior in this group task is to have optimal division of labor between foraging and resting robots as well as optimal allocation of foraging robots on two types of food.

A set of adaptation rules have been extended for individual robot to adapt foraging probability. The individual robot in the group uses only internal cues (from interactions with food source), and social cues (from interactions with other foraging robots) to adapt foraging probability. A desired group behavior of optimal task allocation of foraging robots on two types of food and division of labor between foraging and resting are emerged at group level. With the adaptation mechanism, the design group demonstrates:

- Highly improved foraging performance compared to the control group
- Emergent dynamic allocation of foraging robots on two types of food
- Emergent dynamic division of labor between foraging and resting
- Robustness to environmental disturbance (in food density)

Furthermore, the group with the adaptation mechanism is able to guide the system towards energy optimization despite the limited sensing abilities of the individual robots and the simple behavior rules. The design group also exhibits the capacity to perceive the environment collectively if we take into account how group adapt to the dynamic changes in the environments. That is, more foraging robots of certain type indicate a richer food source of that type. This can only be observed at the overall group level and cannot be deduced from individual robots.

CHAPTER VI

CONTRIBUTIONS AND FUTURE WORK

Swarm robotics provides a new approach to coordinate a large group of simple robots performing a desired group task. Because each robot in the group only has local sense and communication ability, collective behavior is emerged at group level with every robot follows it's own adaptation behavior rule. There is no direct relation between individual behavior and collective behavior, one of challenges in designing swarm robotic system is to understand the effect of individual robot behavior on the collective behavior emerged at group level.

This thesis dedicates the research on design a set of local adaptation rule for a group of robots on collective foraging activity. The goal of the group is to acquire net energy from food source in the environment as fast as possible. When there is one type of food available in the food source, a desired group behavior is an optimal division between foraging and resting among robots so that optimized foraging performance can be achieved.

We have developed a sensor-based computer simulation program for multi robots collective foraging and carried out all the experiments in the simulation program. Behavior based control structure was used to build class module for individual robot. With simple layered basic behavior, each robot is simple to build but also robust to failures.

In order to have a division of labor in the robot group, we adapted response threshold model to determine foraging probability of each robot in the group. Foraging threshold and stimulus were defined. A robot with a higher foraging probability forages more often than a robot with a lower foraging probability. With difference of foraging probabilities among robots, division of labor is emerged at group level.

Since the density of food source in the environment is unknown and possible changing, a set of local adaptation rule has designed for each robot to adapt foraging probability in order to achieve optimal division of labor at group level. The adaptation rule adapts value of foraging threshold and stimulus according to the feedbacks from multiple interactions between foraging robots and environment and between foraging robots.

In the validation experiments, several robot groups with different populations collective forage in a given environment first. At group level, the number of robots actively foraging can reach a dynamic equilibrium at the steady state. That is, a steady division of labor between active foraging and resting could emerge from the local interactions in the design group. The experiment results also show that adaptation rules are able to guide the system towards efficiency optimization.

We have also tested a given robot group with design strategy collective foraging under different environmental conditions - the different growth rate of the food, and observed significantly improved group performance each time, compared with the control group without the adaptation mechanism. The average number of active foraging robots at steady state in a given group varies in different environmental conditions. It can be observed that the richer the food source (with higher food growth rate), the more foraging robots. This implies that the design group has the capacity to collectively perceive the difference of food source in environment. Apparently, it is a new property of individual robots but a emergent capacity at group level.

A step change in the food growth rate is used to generate a dynamic environment. With the adaptation rule, through feedbacks from the multiple interactions between foraging robots and environment, design group could result in a new dynamic equilibrium of division of labor each time, with some time delays. A sudden function loss of some foraging robots is introduced in the experiment to test the robustness of the group. Design group shows robustness by sending some robots that were previous at home out for foraging to recovery the loss.

Proposed adaptation strategy S use both social cue and environment cue to adapt foraging probability. In order to explore how each cue alone affect group behavior, we design strategy S_e and S_r. Strategy S_e only adapts foraging probability from the interactions between foraging robots and food source in the environment. Strategy S_r only adapts foraging probability through information from the interactions between foraging robots.

The experiments shows that when robot group using adaptation strategy S_e, division of labor emerged with slower speed compare to using designed strategy S. When robot group use adaptation strategy S_r, division of labor also emerged but with higher steady state error compare to using strategy *S*. The collective behavior from using strategy S_e depends more on the population of the robot group. The proposed adaptation strategy *S* can guide group to an optimal division of labor both faster and with less steady state error.

After validated the designed adaptation strategy, we extended the adaptation rules to a heterogeneous robot group collective foraging in an environment with two types of food in Chapter 5. The desired group behavior for this task is to have an optimized division of labor between foraging and resting as well as optimized allocation of foraging robots on different type of food.

In first set of experiments, several robot groups with different populations collective forage in a given environment. The experiment results show that the robot group use adaptation strategy is able to guide the system towards efficiency optimization through optimal task allocation and division of labor at group level. The number of robots actively foraging for each type of food reaches a dynamic equilibrium at steady state. That is, a dynamic task allocation between different types of food and division between foraging and resting could emerge from the local multiple interactions between foraging robot and the environment and between foraging robots.

We also test the design group in different environmental conditions - the different growth rate of each type of food, and observed significantly improved performance each time, compared with the control group without the adaptation mechanism. Design group has various task allocations and division of labor at steady state according to the different environment settings.

In a dynamic changing environment, with a step change of food growth rate, adaptation rules could result in a new dynamic equilibrium of task allocation among foraging robots. Design group demonstrate flexibility by adapting to the changes in the environment.

6.1 Summary of Contributions

This research contributes studies in designing local adaptation rules for a group of robots. Collective foraging task is chosen for the robot group and the goal is to achieve optimized energy efficiency. Compared to the related work done by Krieger et al [43], Labella et al [46] and Wenguo et al [35], the contributions of this study can be summarized as follows:

- A new set of interaction rule between foraging robots is designed for local communication of a group foraging robot. Together with interactions between foraging robots and environment, group adapts and achieves the desired collective behavior at faster speed.

- Response threshold function was adapted to regulate the frequency of individual robot foraging activity. With difference of foraging probabilities among robots, division of foraging and resting is emerged at robot group level.

- Collective foraging task was extended to a broader problem which individual robot could behave differently according different type of food in the foraging environment. System changes from simple homogeneous system to more complex heterogeneous system. Through interactions between robot and environment and between foraging robots, division between foraging and resting in robots as well as task allocation of foraging robots on different type of food is emerged at the steady state. As the result, foraging performance of the robot group is greatly improved. Robots that rely on only local sensing and communication exhibit a capacity to perceive information about the food source collectively. The robot group also shows flexibility, robustness to the dynamic environmental changes.

6.2 Future Work

This thesis has investigated the interaction and adaptation rules for collective foraging task in swarm robotics. Although collective behavior of optimal division of labor and task allocation have emerged from multiple interactions between foraging robots and between foraging robots and environment, some questions remain unanswered and need further investigation.

Heterogeneous Robot Group

Although we extended collective foraging to a heterogeneous problem which individual robot could behave differently, there are still a lot possible varieties in term difference on the ability of robot. As collective behavior of task allocation itself, to a great extend, is because the difference among robots. The future work therefore include further investigation of a series of task allocation group behavior according to the task and robot group.

Parameter Validation

In the validation process in this study, we have tested the model with some selected value for adaptation factors. We could not test all the possible value of factors. The future works will therefore include the investigation of an effective selection procedure for testing parameter samples in order to validate the model extensively.

Mathematical Model

The probabilistic model approach has been use in some study of swarm robotics to validate the results in the design. Another extension of this study is to develop a macroscopic probabilistic model for robot group so that the parameter used in the adaptation rules can be optimized.

Finally, because of the practical difficulties, all the experiments in this thesis are implemented in computer simulations and not yet tested in real robots environments. Some issues, for example, robot sensor noise, may arise from the uncertain and noise from real physical environment. The reliability of the adaptation algorithms can only be extensively tested in the real robots environment. Thus, a real robot experiment platform is needed in future work in order to verify both the reliability and results of the adaptation algorithm.

BIBLIOGRAPHY

1. Dorigo, M. and Sahin, E (2004) Swarm Robotics - special issue editorial Autonomous Robots, 17(2-3): pp. 111-113, 2004

2. Brooks, R. A (1986) A robust layered control system for a mobile robot, Journal of Robotic and Automation, vol.2, pp. 14-23, 1986

3. Cao, Y.,Fukunaga, A.,and Kahng, A.(1997). Cooperative mobile robotics: Antecedents and directions. Autonomous Robots, vol.4, pp. 1-23

4. Lynne E. Parker (2000). Current State of the Art in Distributed Robot Systems, Distributed Autonomous Robotic Systems 4, Springer, 2000: pp.3-12

5. Arai, T., Ogata, H., and Suzuki, T.(1989). Collision avoidance among multiple robots using virtual impedance IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 479-485

6. Dudek, G., Jenkin, M., Milios, E., and Wilkes, D (1993) A taxonomy for swarm robots, IEEE/RSJ International Conference on Intelligent Robots and Systems,Yokohama, Japan, pp. 441-447

7. Russell, A.(1993). Mobile Robot Guidance Using a Short-lived Heat Trail Robotics, vol. 11, pp. 427-431

8. Deneubourg, J.-L., Goss, S., Franks, N., Sendova-Franks (1992) The dynamics of collective sorting: robot-like ants and ant-like robots First European Conference on Artificial Life: Toward a Practice of Autonomous Systems, pp. 123-133

9. Melhuish, C., Holland, O. E,, and Hoddell, S (1998) Collective sorting and segregation in robots with minimal sensing, Fifth International Conference on Simulation of Adaptive Behavior: From Animals to Animats V, Zurich, Switzerland

10. Singh, S. P. N,, Thayer, S. M,, and Thayer, W. P.(2002) A foundation for kilorobotic exploration, 2002 Congress on Evolutionary Computation, vol. 2, pp. 1033-1038

11. Pfeifei,R and Scheier, C.(1999) Understanding Intelligence, MIT Press

12. Holland, J.H.(1998) Emergence-From Chaos to Order. Oxford Univ Press.

13. J.L. Deneubourg and S. Goss (1989) Collective patterns and decision-making. Ethology, Ecology and Evolution, 1, pp. 295-311

14. E. Bonabeau, M. Dorigo, and G. Theraulaz (1999) Swarm Intelligence: from natural to artificial systems, Oxford University Press

15. Beni, G. and Wang, J. (1989). Swarm intelligence in cellular robotics systems. In Proceeding of NATO Advanced Workshop on Robots and Biological System

16. Hackwood, S. and Beni, G.(1992) Self-organizing sensors by deterministic annealing.presented at IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1177-1183

17. Deneubourg, J.-L., Theraulaz, G., and Beckers, R.(1992) Swarm-made architectures. presented at European Conference on Artificial Life, pp. 123-133

18. Gamier, S., Gautrais, J., and Theraulaz, G. (2007). The biological principles of swarm intelligence. Swarm Intelligence, 1(1): pp.3-31

19. Dorigo, M. and Birattari, M.(2007) Swarm intelligence. Scholarpedia,2(9): pp. 1462

20. Dorigo, M., Maniezzo, V., and Colorni, (1991) A Positive feedback as a search strategy. Technical Report pp. 91-016

21. Dorigo, M., Maniezzo, V., and Colorni, A.(1996). Ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics - Part B, 26(1): pp. 29-41

22. Kennedy, J. and Eberhart, R.(1995). Particle swarm optimization. In Neural Networks, 1995.Proceedings., IEEE International Conference on, volume 4, pp. 1942-1948

23. Dorigo, M. and Stutzle, T. (2004). Ant Colony Optimization. MIT Press.

24. E.Sahin.(2005). Swarm robotics: From sources of inspiration to domains of application.Swarm Robotics Workshop: State-of-the-art Survey, number 3342

25. Camazine, S., Franks, N. R., Sneyd, J., Bonabeau, E., Deneubourg, J.-L., and Theraulaz, G.(2001). Self-Organization in Biological Systems. Princeton University Press

26. Bayindir, L. and Sahin, E. (2007). A review of studies in swarm robotics. Turkish Journal of Electrical Engineering and Computer Sciences, 15(2): pp. 115-147

27. Beckers, R., Holland, O. E., and Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics. In Brooks, R. A. andMaes, P., editors, Artificial Life IV: Proceedings of the fourth international workshop on the synthesis and simulation of living systems, Cambridge, MA. MIT Press.

28. Matari'c, M. J. and Marjanovic, M. J. (1993). Synthesizing complex behaviors by composing simple primitives. In Proceedings, Self Organization and Life, From Simple Rules to Global Complexity, European Conference on Artificial Life, volume 2, pp.698-707, Brussels, Belgium.

29. Matari'c, M. J. (1994a). Interaction and Intelligent Behavior. PhD thesis, Massachusetts Institute of Technology.

30. Matari'c, M. J. (1995). Designing and understanding adaptive group behavior. Adaptive Behavior, 4(1): pp. 51-80.

31. Sutton, R. S. and Barto, A. G. (1998). Reinforcement Learning: An Introduction. The MIT Press.

32. Matari'c, M. J. (1997). Reinforcement learning in the multi-robot domain. Autonomous Robots,4(1): pp.73-83.

33. Balch, T., Boone, G., Collins, T., Forbes, H., MacKenzie, D., and Santamaria, J. (1995). Ioganymede and callisto: A multiagent robot trash-collecting team. AI Magazine, 16(2): pp. 39-53.

34. Nolfi, S. and Floreano, D. (2000). Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines. Number 0-262-14070-5. MIT Press, Cambridge, MA.

35. Wenguo Liu, Alan F.T. Winfield and Jin Sa. (2007). Modelling Swarm Robotics System: A Case Study in Collective Foraging Task. Proceeding of Towards Autonomous Robotic Systems, pp. 25-32, Aberystwyth, UK. 2007

36. Perez-Uribe, A., Floreano, D., and Keller, L. (2003). Effects of group composition and level of selection in the evolution of cooperation in artificial ants. In Proceedings of the 7th European Conference on Artificial Life ECAL2003, pp. 128-137.

37. Tarapore, D., Floreano, D., and Keller, L. (2006). Influence of the level of polyandry and genetic architecture on division of labour. In Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems, pp.358-364.

38. Fontan, M. S. and Mataric, M. J. (1996). A study of territoriality: The role of critical mass in adaptive task division. In From Animals to Animats 4, Fourth International Conference on Simulation of Adaptive Behavior (SAB-96), pp. 553-561.

39. stergaard, E. H., Sukhatme, G. S., and Mataric, M. J. (2001). Emergent bucket brigading a simple mechanism for improving performance in multi-robot constrained-space foraging tasks. In Proceedings of the 5th International Conference on Autonomous Agents, pp. 2930, New York, NY, USA. ACM Press.

40. Lerman, K. (2002). Mathematical model of foraging in a group of robots: Effect of interference. Autonomous Robots, 13(2): pp. 127-141.

41. Sugawara, K., Kazama, T., and Watanabe, T. (2004). Foraging behavior of interacting robots with virtual pheromone. In Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, volume 3, pp. 3074-3079 vol.3.

42. Theraulaz, G., Bonabeau, E., and Deneubourg, J.-L. (1998). Response threshold reinforcements and division of labour in insect societies. In Royal Society of London.

43. Krieger, M. J. B. and Billeter, J.-B. (2000). The call of duty: Self-organized task allocation in a population of up to twelve mobile robots. Robotics and Autonomous Systems, 30(1-2): pp. 65-84.

44. Krieger, M. J. B., Billeter, J.-B., and Keller, L. (2000). Ant-like task allocation and recruitment in cooperative robots. Nature, pp. 992-995.

45. Labella, T. H., Dorigo, M., and Deneubourg, J.-L. (2004). Efficiency and task allocation in prey retrieval. In Proceedings of the First International Workshop on Biologically Inspired Approaches to Advanced Information Technology, volume 3141, pp. 32-47.

46. Labella, T. H., Dorigo, M., and Deneubourg, J.-L. (2006). Division of labor in a group of robots inspired by ants foraging behavior. ACM Transactions on Autonomous and Adaptive Systems, 1(1): pp. 4-25.

47. Jones, C. and Matari'c, M. J. (2003). Adaptive division of labor in large-scale multi-robot systems. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1969-1974, Las Vegas, Nevada.

48. Guerrero, J. and Oliver, G. (2003). Multi-robot task allocation strategies using auction-like mechanisms. Artificial Research and Development in Frontiers in Artificial Intelligence and Applications, 100: pp. 111-122.

49. Winfield, A. F. T. (2008). Foraging robos. Springer Encyclopaedia of Complexity and System Science.

50. Camazine, S., Franks, N. R., Sneyd, J., Bonabeau, E., Deneubourg, J.-L., and Th'eraulaz, G. (2001). Self-Organization in Biological Systems. Princeton University Press.

51. Kube, C. R. and Zhang, H. (1993). Collective robot intelligence. In Proceedings of the second international conference on From animals to animats 2 : simulation of adaptive behavior, pp. 460-468, Cambridge, MA, USA. MIT Press.

52. Melhuish, C., Holland, O., and Hoddell, S. (1999). Convoying: using chorusing to form travelling groups of minimal agents. Robotics and Autonomous Systems, 28(2-3): pp. 207-216.

53. Trianni, V., Gro, R., Labella, T., S ahin, E„ and Dorigo, M. (2003). Evolving Aggregation Behaviors in a Swarm of Robots. In Banzhaf,W., Christaller, T., Dittrich, P., Kim, J. T., and Ziegler, J., editors, Advances in Artificial Life - Proceedings of the 7th European Conference on Artificial Life (ECAL), volume 2801 of Lecture Notes in Artificial Intelligence, pp. 865-874. Springer Verlag, Heidelberg, Germany.

54. Dorigo, M., Trianni, V., Sahin, E., Gross, R., Labella, T. H., Baldassarre, G., Nolfi, S.,Deneubourg, J.-L., Mondada, F., Floreano, D., and Gambardella, L. (2004). Evolving selforganizing behaviors for a swarm-bot. Autonomous Robots, 17: pp. 223-245.

55. Bahgeci, E. and Sahin, E. (2005). Evolving aggregation behaviors for swarm robotic systems: a systematic case study. In Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE, pp. 333-340.

56. Soysal, O. and Sahin, E. (2005). Probabilistic aggregation strategies in swarm robotic systems. In Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE, pp. 325-332.

57. Gamier, S., Jost, C., Jeanson, R., Gautrais, J., Asadpour, M., Caprari, G., and Theraulaz, G.(2005). Collective decision-making by a group of cockroach-like robots. In Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE, pp. 233-240.

58. Jeanson, R., Rivault, C., Deneubourg, J. L., Blanco, S., Fournier, R., Jost, C., and Theraulaz, G. (2005). Self-organized aggregation in cockroaches. Animal Behaviour, 69(1): pp. 169-180.

59. Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. In SIGGRAPH 87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques, pages 25-34, New York, NY, USA. ACM Press.

60. Brogan, D. C. and Hodgins, J. K. (1997). Group behaviors for systems with significant dynamics. AutonomousRobots, 4: pp. 137-153.

61. Toner, J. and Tu, Y. (1998). Flocks, herds, and schools: A quantitative theory of flocking. Physical Review E„ 58(4): pp. 4828-4858.

62. Yamaguchi, H. and Beni, G. (1996). Distributed autonomous formation control of mobile robot groups by swarm-based pattern generation. In Int. Symp. on Distributed Autonomous Robotic Systems DARS-96.

63. Shimoyama, N., Sugawara, K., Mizuguchi, T., Hayakawa, Y., and Sano, M. (1996). Collective motion in a system of motile elements. Physical Review Letters, 76(20): pp. 3870-3873.

64. Tanner, H., Jadbabaie, A., and Pappas, G. (2003). Stable flocking of mobile agents, part i: fixed topology. In Decision and Control, 2003. Proceedings. 42nd IEEE Conference on, volume 2, pp. 2010-2015 Vol.2.

65. Li, X., Xiao, J., and Cai, Z. (2005). Stable flocking of swarms using local information. In Systems, Man and Cybernetics, 2005 IEEE International Conference on, volume 4, pp. 3921-3926 Vol. 4.

66. Kelly, I. D. and Keating, D. A. (1996). Flocking by the fusion of sonar and active infrared sensors on physical autonomous mobile robots. In The Third International Conference on Mechatronics and Machine Vision in Practice, volume 1, pp. 1-4, Guimaraes, Portugal.

67. Brooks, R. A. (1986). A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, 2(1): 1423.

68. Vaughan, R., Sumpter, N., Henderson, J., Frost, A., and Cameron, S. (1998). Robot control of animal flocks. In Intelligent Control (ISIC), 1998. Held jointly with IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA), Intelligent Systems and Semiotics (ISAS), Proceedings of the 1998 IEEE International Symposium on, pp. 277-282.

69. Hanada, Y., Lee, G., and Chong, N. Y. (2007). Adaptive flocking of a swarm of robots based on local interactions. In Swarm Intelligence Symposium, 2007. SIS 2007. IEEE, pp. 340-347.

70. Nembrini, J. (2005). Minimalist Coherent Swarming of Wireless Networked Autonomous Mobile Robots. PhD thesis, Univeristy of the West of England, Bristol.

71. Bjerknes, J., Winfield, A., and Melhuish, C. (2007a). An analysis of emergent taxis in a wireless connected swarm of mobile robots. In Swarm Intelligence Symposium, 2007. IEEE, pages 45.

72. Deneubourg, J.-L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., and Chretien, L. (1991). The dynamics of collective sorting: Robot-like ants and ant-like robots. In Meyer, J.-A. and Wilson, S. W., editors, Proceedings of the First International Conference on Simulation of Adaptive Behavior: From Animals to Animats, pp. 356-363. MIT.

73. Beckers, R., Holland, O. E., and Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics. In Brooks, R. A. andMaes, P., editors, Artificial Life IV: Proceedings of the fourth international workshop on the synthesis and simulation of living systems,Cambridge, MA. MIT Press.

74. Martinoli, A., Franzi, E., and Matthey, O. (1998). Towards a reliable set-up for bio-inspired collective experiments with real robots. In he Fifth International Symposium on Experimental Robotics V, pp. 597-608, London, UK. Springer-Verlag.

75. Melhuish, C., Wilson, M., and Sendova-Franks, A. (2001). Patch sorting: Multi-object clustering using minimalist robots. In ECAL 01: Proceedings of the 6th European Conference on Advances in Artificial Life, pp. 543-552, London, UK. Springer-Verlag.

76. Wilson, M.,Melhuish, C., Sendova-Franks, A. B„ and Scholes, S. (2004). Algorithms for building annular structures with minimalist robots inspired by brood sorting in ant colonies. Auton. Robots, 17(2-3): pp. 115-136.

77. Wang, T. and Zhang, H. (2004). Collective sorting with multiple robots. In Proceedings of the First IEEE International Conference on Robotics and Biomimetics.

78. Verret, S., Zhang, H., and Meng, M.-H. (2004). Collective sorting with local communication. In Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, volume 3, pp. 2687-2692 vol. 3.

79. Parker, L. E. (1994). Alliance: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots. In Proceedings of the 1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems, pp. 776-783.

80. Jennings, J., Whelan, G., and Evans, W. (1997). Cooperative search and rescue with a team of mobile robots. In Advanced Robotics, 1997. ICAR 97. Proceedings., 8th International Conference on, pp. 193-200.

81. Reif, J. H. and Wang, H. (1999). Social potential fields: a distributed behavioral control for autonomous robots. Robotics and Autonomous Systems, 27(3): pp. 171-194.

82. Dudenhoeffer, D. D. and Jones, M. P. (2000). A formation behavior for large-scale micro-robot force deployment. In Proceedings of the 32nd conference on Winter simulation.

83. Dudenhoeffer, D. D., Bruemmer, D. J., Anderson, M. O., and McKay, M. D. (2001). Development and implementation of large-scale micro-robotic forces using formation behaviors. Volume 4364, pp. 159-168. SPIE.

84. Bruemmer, D. J., Dudenhoeffer, D. D., and Marble, J. L. (2001). Mixed-initiative remote characterization using a distributed team of small robot. In 2001 AAAI Mobile Robot Workshop, Seattle, WA.

85. Bruemmer, D. J., Dudenhoeffer, D. D., McKay, M. D., and Anderson, M. O. (2002). A roboticswarm for spill finding and perimeter formation. In Spectrum, 2002.

86. Hayes, A. T., Martinoli, A., and Goodman, R. M. (2000). Comparing distributed exploration strategies with simulated and real autonomous robots. In Parker, L. E.,

Bekey, G., and Bahren, J., editors, Proceeding of the fifth International Symposium on Distributed Autonomous Robotic System, pp. 261-270, New York. Springer.

87. Hayes, A. T., Martinoli, A., and Goodman, R. M. (2002). Distributed Odor Source Localization. IEEE Sensors Journal, 2(3): pp. 260-271. Special Issue on Artificial Olfaction, H. T. Nagle, J. W. Gardner, and K. Persaud, editors.

88. Hayes, A. T., Martinoli, A., and Goodman, R. M. (2003). Swarm Robotic Odor Localization: Off-Line Optimization and Validation with Real Robots. Robotica, 21(4): pp. 427-441. Special issue on Biological Robotics, D. McFarland, editor.

89. Correll, N. and Martinoli, A. (2005a). Collective inspection of regular structures using a swarm of miniature robots. In Springer Tracts in Advanced Robotics, volume 21.

90. Caprari, G. and Siegwart, R. (2003). Design and control of the mobile micro robot alice. In Proceedings of the 2nd International Symposium on Autonomous Minirobots for Research and Edutainment, pp. 23-32, Brisbane, Australia.

91. Pugh, J. and Martinoli, A. (2007). The cost of reality: Effects of real-world factors on multi-robot search. In Robotics and Automation, 2007 IEEE International Conference on, pp. 397-404.

92. Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In Neural Networks, 1995. Proceedings., IEEE International Conference on, volume 4, pp. 1942-1948 vol.4.

93. Stilwell, D. J. and Bay, J. S. (1993). Toward the development of a material transport system using swarms of ant-like robots. In International Conference on Robotics and Automation, pp. 766-771.

94. Parker, L. E. (1999). Adaptive heterogeneous multi-robot teams. Neural Networks and Their Applications, 25: pp. 72-95.

95. Yamada, S. and Saito, J. (1999). Adaptive action selection without explicit communicationfor multi-robot box-pushing. In 1999 IEEE/RSJ International Conference on Intelliegent Robots and Systems, pp. 1444-1449.

96. Kube, C. R. and Bonabeau, E. (2000). Cooperative transport by ants and robots. Robotics and Autonomous Systems, 30(1-2): pp. 85-101.

97. Gross, R. and Dorigo, M. (2004a). Coperative transport of objects of different shapes and sizes. In Ant Colony Optimization and Swarm Intelligence, 4th International Workshop, volume 3172, pp. 107-118.

98. Gross, R. and Dorigo, M. (2004b). Evolving a cooperative transport behavior for two simple robots. In Artificial Evolution-6th International Conference, volume 2936, pp. 305-317.

99. Gross, R. and Dorigo, M. (2004c). Group transport of an object to a target that only some group members may sense. In Parallel Problem Solving from Nature-8th International Conference PPSN VIII, volume 3242, pp. 852-861.

100. Ijspeert, A. J., Martinoli, A., Billard, A., and Gambardella, L. M. (2001). Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment. Autonomous Robots, 11(2): pp. 149-171.

101. Martinoli, A. (1999). Swarm Intelligence in Autonomous Collective Robotics: From Tools to the Analysis and Synthesis of Distributed Collective Strategies. Ph.d. thesis, Ecole Polytechnique Federale de Lausanne.

102. Agassounon, W. and Martinoli, A. (2002b). A Macroscopic Model of an Aggregation Experiment using Embodied Agents in Groups of Time-Varying Sizes. In Proc. of the IEEE Conf. on System, Man and Cybernetics SMC-02.

103. Correll, N. and Martinoli, A. (2005a). Collective inspection of regular structures using a swarm of miniature robots. In Springer Tracts in Advanced Robotics, volume 21.

104. Correll, N. and Martinoli, A. (2007a). Modeling Self-Organized Aggregation in a Swarm of Miniature Robots. In IEEE 2007 International Conference on Robotics and Automation Workshop on Collective Behaviors inspired by Biological and Biochemical Systems.

105. Robinson, G. E. (1987). Modulation of alarm pheromone perception in the honeybee: evidence for division of labor based on hormonally regulated response thresholds. J. Comp. Physiol. A 160, pp. 613-619.

106. Robinson, G. E. (1992). Regulation of division of labor in insect societies. A. Rev. Entomol. 37, pp. 637-665.

107. Plowright, R. C. Plowright, C. M. S. (1988). Elitism in social insects: a positive feedback model. In Interindividual behavioral variability in social insects (ed. R. L. Jeanne), pp. 419-431. Boulder, CO: Westview Press.

108. Bonabeau, E., Theraulaz, G. Deneubourg, J.-L. (1996). Quantitative study of the fixed threshold model for the regulation of division of labor in insect societies. Proc. R. Soc. Lond. B 263, pp. 1565-1569.

109. Bonabeau, E., Theraulaz, G. Deneubourg, J.-L. (1998). Fixed response thresholds and the regulation of division of labour in insect societies. Bull. Math. Biol.

110. Martinoli and F. Mondada, (1998). Probabilistic Modelling of a Bio-Inspired Collective Experiment with Real Robots, In Proceeding of the Third International Symposium on Distributed Autonomous Robotic Systems, pp. 25-27, MIT Press

111. K. Lerman and Aram Galstyan (2001). A General Methodology for Mathematical Analysis of Multi-Agent Systems, USC Information Sciences Technical Report ISI-TR-529