TOWARD A SYNTHETIC ECOSYSTEM: GROUP BEHAVIOR OF AUTONOMOUS MOBILE ROBOTS.

By
Scott Jantz

A THESIS PRESENTED TO THE GRADUATE SCHOOL OF
THE UNIVERSITY OF FLORIDA IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF ENGINEERING

UNIVERISTY OF FLORIDA

1998

Abstract of Thesis Presented to the Graduate School of the University of Florida in Partial Fulfillment of the Requirements for the Degree Master of Engineering

TOWARD A SYNTHETIC ECOSYSTEM: GROUP BEHAVIOR OF AUTONOMOUS MOBILE ROBOTS.

By
Scott Jantz

May, 1998

Chairman: Professor Keith L. Doty
Major Department: Department of Electrical and Computer Engineering

This thesis covers a series of experiments using autonomous mobile robots designed to explore the role of low level behaviors in creating emergent complexity. Emergent behavior has profound implications not only in biology and robotics but also engineering and education along with sociology and psychology. The experiments in this thesis include large swarm experiments with TJ robots and long duration experiments with recharging robots. The use of robotics in education is also explored. The ideas developed through swarm robotics are brought into a larger context through their inclusion in a new education paradigm.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Introduction

The purpose of this thesis is to explore the role of autonomous mobile robots as participants in a new biology, one that Kelly (1994), describes as the marriage of the made and of the born. The “born” refers to all “living”, natural creatures (including humans), and the “made” are those creatures built by humans. Since the ideas in this thesis cut across many disciplines the next section defines a number of terms and their usage here.

Definitions

Autonomous mobile robots. The word robot comes from a Czechoslovakian word meaning “forced servant,” first coined in the play R.U.R. (Rossum’s Universal Robots) by the writer Karel Capek in 1920. The term autonomous refers to the independent control of each robot by its on-board computer. Finally, the term mobile refers to the ability of the robots to maneuver around an environment, and, in the case of all the robots in this thesis, untethered and self-reliant.

Convergent and divergent engineering. These terms are used to describe a new way of thinking which stems from the idea of synthetic ecosystems. Convergent engineering is engineering which takes advantage of the emergent group behavior of components to produce a more biological and resilient creation. Divergent engineering is the current engineering where the emergent or unexpected behavior of components is undesired and can lead to catastrophic failures because all possible interactions cannot be tested. Convergent engineering brings into play the interactions between pieces of a whole; divergent engineering focuses on the pieces themselves often working against their interactions with each other.

Ecosystem. A community of organisms together with their environment, functioning as a whole. In this research, the ecosystem is populated with various mobile robots and humans. Environments range from the simple, square-walled empty arenas, to the very complex, rooms filled with complex furniture and young children. The key concept here is the idea of organisms and an environment interacting in a way which produces complex behavior. The gestalt is important; the system is more complex than its parts.

Group behavior and emergent behavior. Behavior is an agents response to stimuli. The term group behavior refers to the complex interactions of simple individuals to each other and their environment. Emergent behavior refers to behaviors that arise out of the interactions of behaviors and the environment that are not expressly coded into the agent. Again, the gestalt is key. Although the individuals behavior may be simple, interactions amongst individuals often produce complex *emergent* behavior. This is a surprisingly common observation. The behavior of the same substance under different conditions and scales can change radically. Often the phenomena appear to have nothing in common. For example, steam coming out of a pot of boiling water behaves very differently than a thundercloud. Both the steam and the thundercloud consist of the same material, water vapor, but in different scales and in different environments.

Synthetic. Not natural in origin, man-made. In the context of this paper the robots are artificial, but the ecosystem that they exist in is also man-made to an extent. Although a human artifact, the ecosystem is not controlled in an absolute way by the creators. This lack of absolute control is a major point in this thesis.

A major goal of this thesis is to explore the decentralized, self-organizing nature of swarm robots. The problems of self-organization, synthetic ecosystem design and group behavior span diverse, and inherently complex issues, requiring novel approaches. Instead of attacking this large problem directly, I will nibble at the edges with a series of elucidating experiments that begin to shed light on a subject that is oxymoronically, both ubiquitous and elusive.

The rules of self-organization are at work in every facet of our lives, from the cells and tissues in our bodies to global weather patterns or to the interactions of parts in a modern aircraft. As pervasive as the effects of these natural laws are, they have been largely ignored by modern science because they are so difficult to quantify. Traditional engineering and science have simplified and smoothed over most of these complex interactions with first order approximations. Physical chemistry tells us much about the molecular motion of individual particles but offers little insight into weather patterns, where rules of thumb and statistical techniques take over. There exists no encompassing theory to unite these two phenomenon, of molecular motion and weather patterns into a central theory. So large and culturally ingrained is this disparity that even the organization of our fields of study reflect it. Study a relatively small number of air molecules, and you are a chemist. Study huge numbers of these same molecules, and you become a meteorologist with tools and techniques which seem to have little in common with chemistry. Likewise, biologists can tell us much about the behavior of one individual living organism, but encounter difficulties in more complex colonies and ecosystems. I cannot claim to have broken through these barriers in this thesis, but my experiments in this area should introduce novel ways of approaching the problem. Ultimately, perhaps, new ways of engineering a world that has both the advantages and resiliency of an ecosystem with the accuracy and purpose of an engineered machine will emerge.

Grazer: The struggle to survive

Darwin introduced the world to the power of natural selection and competition. As a fitting tribute to him, the first experiments I describe will center around a rather blood-thirsty yet extremely virulent species of robots misnamed Grazer. These robots compete for “food” (a single recharge station) in which success means an extended life and failure means death. My studies chronicle autonomous, untethered experiments for more than a week in duration, some of the longest run-times for fully autonomous robots to date. The robots were actively moving 80% of the time. Mark Tilden has solar robots that move very little but have been technically active for years (Reitman, 1984).

TJ: The swarm develops

Moving on to larger, although shorter duration experiments, I present the mass-produceable robots know as TJ[1] (for Talrik Junior, the smaller version of the larger robot Talrik) to explore the dynamics of large groups. In these experiments I introduce some novel approaches for the quantification of robot group performance, a lacking structure in the world of robotics. Experimenting with groups of up to 30 robots, another barrier has been broken with the world's largest, working swarm. Mataric (1991) has a smaller swarm with about 25 robots, although not all of them work at the same time. My experiments yielded some interesting results which could increase our understanding of the link between pure chemical interactions and the complexities of life.

TJ robots: Birds of a feather

Exploring quasi-coordinated activity with smaller groups of TJ robots, the third set of experiments, shows how the mindset must change when trying to get a group of robots to self organize. In these experiments, the robot's behavior in clusters may have theoretical bearings on the action of colloidal systems (Atkins, 1994). These experiments also show how emergent behavior can be used to simplify an algorithm and produce a more robust behavior at the same time.

[1] Talrik, TJ, Talrik Junior and TJ PRO are all trademarks of Mekatronix Corp.

TJ robots in education

In the final experiments, I take the lead from Mitchel Resnick (1994) by taking large groups of robots into elementary and middle school classrooms in a series of guided activities where students learn about autonomous robots. Here I explore several methods of how to teach robotics, math and science along with an introduction to the ideas in this thesis. I taught the ideas of decentralization, distributed control and complexities in group interactions to middle school students along with the idea of emergent behavior. I did not use technical jargon but introduced them to the students in a way that enabled them to recognize and approach these unsolved problems hopefully as future engineers and scientists. This educational outreach work involving the Machine Intelligence Lab is still being developed and therefore I lack long term studies on the potential educational impact of our work. Since much of science has reduced the world to sets of linear equations, the true breakthroughs in the future will require new, ways of thinking. The students I worked with are developing this kind of thinking by working with our robots.

Swarm robotics

Swarm robotics is not simply many robots working together. A modern automotive factory floor with dozens of industrial manipulators would erroneously be labeled a "swarm" if that were so. "Swarm" in this research is characterized by decentralized control and self-organization an approach in opposition to the central control paradigm that is widespread in modern engineering. In a modern automotive factory, a computer precisely controls every action and there is very little local independent control. Central control allows the factory to operate with a high degree of efficiency. The downside of this approach is that these centrally controlled configurations are both inflexible and fragile, traits not commonly found in a biological system. Biological systems tend to be sub-optimal in efficiency, but very resilient and flexible when perturbed. Besides high efficiency the main reason that the central control paradigm has been so prevalent in modern engineering is that it can be rigorously controlled and analyzed. As manufacturing moves toward greater complexity, and requires increasing robustness and flexibility, our solutions may begin to approach the order of biological complexity. At that point the central control paradigm will need to be replaced.

Another application of this shift in thinking is in the analysis of economies and the governments that run them. We can already find one example of this complexity in the world economy. International trade contributes to an economy that is so complex that central control by the state is impossible. The inability to manage a complex economy from a central control perspective was evident in the Soviet Union before its collapse or *decentralization* (Kaufman, 1995). To apply decentralized control in engineering, a new engineering paradigm is needed, along with the tools of analysis and design tailored to this different style of thinking.

Biological background

One of the more profound ethological discoveries in the animal kingdom is that the queen in an ant society is *not* the instrument of central control that was once thought. Although the queen emits regulating pheromones, she does not by any means control every aspect of the colony (Holldobler and Wilson, 1990). Even without this central control the colony manages to do seemingly impossible acts: construct elaborate colonies designed with cooling vents and drainage ditches, care for young, enslave other ant species, search for and retrieve food in raiding parties, and grow and harvest crops. What can we draw upon from these solutions produced by billions of years of evolution which have allowed the ant species to succeed? The idea that engineered creations can draw upon biological solutions and indeed can become a living entity in some aspects, is known as artificial life (ALIFE). One goal of ALIFE is to draw upon the characteristics of living systems to engineer intelligent creations while simultaneously learning more about real (or wet) life (Levy, 1992).

CHAPTER 2

REVIEW OF LITERATURE

Background research

Wet life

The place to start the development of artificial organisms and the study of their interactions is with our best and only concrete example: living organisms. From the natural sciences, we can gain insight into the key characteristics of life and the underlying principles which separate the animate from the inanimate.

Life is defined in Webster's (1997) *New Universal Unabridged Dictionary* as "that property of plants and animals that makes it possible for them to take in food, take energy from it, grow, adapt themselves to their surroundings, and reproduce their kind." Automata are defined by Von Neumann as self-operating machines (Levy, 1992). These machines can be natural or artificial. The term "wet life" refers to all natural living things which depend on water and carbon to survive (Levy, 1992).

Physical laws and wet life

The foundations of life intertwine with the components of life, molecules, and the rules that govern their interactions, i.e., chemistry. How simple components organize themselves into the complex building blocks of life has profound implications for the macroscopic behavior of living systems. Kauffman (1995) believes that the same laws that govern microscopic interactions also drive the macroscopic complexities seen in organisms, ecologies, societies and economies. For example, the pheromones that ants use to lay trails are linked to the physical capability of volatile organic compounds to evaporate and disperse over time. This volatility allows ants to determine the age of trails and to follow odor gradients to stay on track (Holldobler and Wilson, 1990)(Resnick, 1994). Another linkage between the macroscopic and microscopic world is the physiological behavior of habituation. Habituation takes place in all animals as an animal ignores a repeated stimulus over time. Habituation is linked both theoretically and physiologically by the processes of chemical equilibrium and shifting of products to reactants. In a nerve cell this repeated stimulation shifts the equilibrium toward products until the reactants are temporally depleted, resulting in a habituation period. Behavior has its roots in chemistry the link being the neurobiology of the individual (Carlson, 1994).

Self-organization

The principle of self-organization plays a crucial part in the development and continuation of life. Whether leading to army ant columns or the complex temperature regulating structures in termite mounds, the principles at work are the same (Howse, 1970). The control in both of these instances is local and individual; the insect uses only local stimuli and its own internal behaviors to construct extremely complex structures. The resulting structure is encoded not from some grand blueprint but instead from the interactions between many individuals and the environment. One problem in studying these phenomena is that the only way to determine what structure is going to result is to run the system. Each individual insect needs only its innate behaviors and not the memorized blueprint of the structure, greatly reducing the amount of brain power needed by the insects. Because of the interchangeability of the individual insects and the distributed nature of the task, if half of the individuals were killed the task would still proceed except at a slower pace. A similar process is at work in embryology, where a hand full of genes directs the structure of an entire organism. The key to understanding how self-organization works is to first understand how nature uses self organization through animal behavior.

Animal behaviors

The heart of natural self-organizational systems is the animal behavior from which these systems are created. Even though societies of ants are extremely complex and the species is capable of great acts, the actual number of behaviors is astoundingly small. A total of 27 different behaviors make up the repertoire of ants, with workers only exhibiting eight or nine of these (Holldobler and Wilson, 1990). These numbers reveal the high data efficiency that self-organization yields. Another example of this extreme data compression is the human brain. If every base pair in the human genome where devoted to connecting up the human brain (clearly an exaggeration), then to connect the estimated 100 billion cells with the average 5,000 connections each base pair would produce 250,000 connections (Carlson, 1994). This leads to a conservative 250,000:1 compression ratio, one far beyond any known compression algorithm.

The word "behavior" has come to mean many things to many people. Brooks (1990) uses it to mean an action in response to stimuli or remembered stimuli. This definition agrees with the behavioral psychologist view. Key to the definition is the idea of action. The staunch behavioral psychologist claims that one only need look at the external actions of an organism to completely understand its behavior. I adopt this action-based definition with respect to robots in this thesis. The other problematic question is what is an atomic behavior and what is a compound behavior. Although this is not as important to ethologists, this point is extremely important to roboticists who must program behaviors. For example, the circling behavior that ants exhibit when laying an alarm pheromone (Holldobler and Wilson, 1990) seems to be a compound behavior. In my simulations of ants, the atomic behaviors "drop alarm pheromone if disturbed" and "follow alarm trail" combine to the compound behavior of the ant following its own trail. Herein lies the key to several important points in this thesis. One, atomic behaviors combine to produce compound behaviors. Secondly, the resultant compound behavior is *emergent*, one that is not expressly programmed but rather the result of atomic behavior interaction coupled with the environment or another individual's behavior. Nowhere in the code for the simulated ant is a "circle while laying pheromone" command. Yet while observing the ant, an ethologist would write down circling behavior as part of the repertoire of the ant. Determination of atomic behaviors demostrates what an important tool ALIFE can be to biologists. The discovery of truly atomic behaviors may be impossible without artificial creatures on which to experiment. Artificial organisms allow total control to determine which programmed behaviors lead to resulting actions. Ethology is an important tool in the examination of behaviors and their effects in robots.

Ethology

Ethology is the study of animal behavior. From this biological science we can learn several important things that will help in the development of robots and the analysis of their performance. First is the concept of the ethogram, a complete description of the behavioral repertoire of an individual or species with a specification of the frequency of acts and the transitional probabilities between them (Holldobler and Wilson, 1990). This technique is akin to reverse engineering an animal. The ethogram is the resulting combination of behaviors of the animal (or robot) and the environment. A programmed (or evolved) behavior may never appear in the ethogram if the environment does not trigger it. Likewise, an ethologist may record a behavior in the ethogram that is not programmed into the individual organism. This would be an emergent behavior, the result of the interaction of the environment and the other behaviors of the organism. Ethograms resemble state transition graphs from finite state machines, leading us to the question: Is there any fundamental difference at the abstract level between ALIFE and natural life?

Artificial life

Definitions

Chris Langton originally coined the term Artificial life (or ALIFE) to cover a range of ideas concerned with attempts to synthesize phenomena normally associated with natural living systems (Levy, 1992). The media for these synthesis experiments include computers, robots and chemical soups. The nature of this nascent field is to understand life by creating man-made life.

Physical laws and ALIFE

There is a strong theoretical link between the physical laws that govern particles and the foundations of natural or man-made life. This link is illuminated by Kauffman (1995) when he examines the relationship between self-organizing molecules and the formation of life on Earth. These same laws apply to ALIFE, where simple reaction laws dictate how cellular automata (CA) behave. From the simple rules governing the interaction of these simulated molecules spring surprising life-like complexity. From the extensive work done in the ALIFE field with computer simulations, simple molecular interactions seem to be a promising area of exploration for robotics. I based many of my experiments on this work in ALIFE. A principle conjecture of this thesis is that low level physical interactions lead to more complex behaviors in robots. Experiments with robots whose behavior is governed by physically based interactions may also lead to a more fundamental understanding of what abilities are required of a robot to produce more complex behaviors (Steels, 1994) .

Emergent behaviors

Emergent behavior is an integral part of the theory of artificial life. The proposition that simple interactions among agents lead to complex unexpected behaviors is an integral part of ALIFE. Given simple rules, the emergent behavior of some ALIFE programs has been to create intricate hive-like structures. The design of these structures is not implicitly programmed into the agents but rather a result of the interactions of simple rules in multiple agents (Resnick, 1994) (Kelly, 1994) (Levy, 1992). Self-generating CAs have been created with the emergent behavior being the creation of other CAs, resulting in a digital embryology (Levy, 1992).

Learning

The robot learning field is large and clearly beyond the scope of this thesis. Several good references discuss the area of robot learning (Hexmoor and Nute, 1992) (Jaakkola, Jordan and Singh, 1993) (Mahadevan and Connell, 1992) (Mahadevan, 1994). While learning is clearly a promising area for developing robot behaviors, there are still some shortcomings. My purpose here is not to develop a novel learning strategy, but rather to show how groups of robots can aid in overcoming some of the problems associated with learning by real robots. One of the issues in learning is the large time scales needed to converge to an optimum strategy (many times these runs can only be done in computer simulations (Terzopoulos, Tu and Grzeszczuk, 1994) because real robots currently cannot operate long enough for their learning algorithms to converge). My work with the Grazer robots begins to address this problem.

Computer simulations and tools

Although simulations of robots miss many of the subtle interactions between environment and internal robot behavior that result in emergent behavior, simulation is still useful. Simulations have a place in exploring experiments that are simply too complex, costly or time consuming to be done with real robots. Simulations give insight into how algorithms on robots may act out in large groups or over long periods of time. Another interesting use for simulations involves the direct comparison of simulation to real robot experiments. One of the non-qualitative results that I obtained while working with simulations was the ability of simulated robots to get “stuck.” A very simple algorithm in a real robot will almost never become trapped. Even if the algorithm has a flaw that will cause the robot to oscillate in certain environments, the noise in the environment coupled with the inaccuracies in actuation will make it virtually impossible for the robot to get stuck. If a simulated agent starts oscillating, it will become trapped because it moves exactly the same distance each time and sees the same sensor readings. Environmental randomness is very difficult to predict and even harder to model. In some cases, the randomness which allows a robot running a particular algorithm to navigate an environment successfully is only evident after the experiment has been performed. A connection exists between environmental noise and randomness, and between robot action and chaos theory. For this reason, the only way to test an algorithm on a real robot is to run it (similar to chaotic systems) and observe the outcome. Problems with accurate simulation have lead us to quip that “Simulation is hard”. This is not to say that simulation is worthless. It provides great insight into experiments that may be impossible or difficult to perform. Comparing simulation with real robots allows us to observe how a real environment influences the behavior algorithm. The performance of some algorithms differ little in simulation and reality (as seen later in the effusion experiments) and the performance of others completely fall apart in simulation (as do certain collision avoidance algorithms). The simulation tool used in this thesis is Star-logo running on a Power Mac (Resnick, 1994).

Robotic ALIFE

Simulations of ALIFE and studies of natural life have lead researchers to build robots that have ALIFE characteristics: they can survive in an environment, adapt to changes and even learn and evolve. Walter (1950) performed some of the first work with ALIFE robots. His simple analog vacuum tube robots could recharge themselves. In the 1990's Brooks (1990) at MIT initiated a decidedly ALIFE approach (as opposed to the traditional AI approach). Brook's robots would react to their environment much like simple animals instead of logically reasoning their way through the environment like the traditional AI approach (Brooks, 1990). Brooks' approach was to some extent inspired by the mind-game book that Valentino Braitenburg wrote called *Vehicles*. In this book Braitenburg (1984) dreams up simple reactive creatures whose actions directly couple with their perception of the environment. Some of the research that has come out of the University of Florida's Machine Intelligence Lab (MIL) has been based on Brooks and Braitenburg (Doty and Harrison, 1993) (Doty and Seed, 1994) (Doty and Van Aken, 1993) (Caselli et. al., 1994). From singular robots, researchers then moved to the fascinating area of group or swarm robotics, where large groups of robots interact like an insect hive. The driving ideas behind swarm robotics are based on simulation with insect-like creatures which are able to construct complex hives and perform group interactions like flocking and other decentralized behaviors. Self-organization is perhaps the only way to coordinate large groups of robots since both the environmental uncertainties and the complexity in coordinating every movement of the robots would be very susceptible to perturbation. To find a solution to this problem, researchers started looking at groups of robots and how they can interact to produce self-organizing behaviors (Mataric, 1992) (Mataric, 1994) (Parker, 1994) (Parker, 1995) (Parker, 1993) (Wang and Premvuti, 1994) (Wang et. al., 1994) (Yagi et. al., 1994). In her dissertation Mataric (1994) works with a group of 20 robots and is the first person to do work of this scale with real robots. One shortcoming of her research is the reliance on global knowledge.

Because all the robots know where they are and where the other robots are all the time, they do not need to work with local knowledge. Another problem which stems from this approach is that the behaviors on the robots are expressly programmed or learned. There is little reliance on emergent behavior. Her approach makes the behaviors of the robots more fragile since expressly programmed behaviors fail when unanticipated situations arrise. Letting the robots and environment assemble more complex robot behaviors, from the most simple or atomic level, is the central theme to all of the following experiments. Discovering these atomic behaviors and learning how to assemble them into more complex emergent behaviors will give us insight into animal behavior and make robots more versatile and reliable in unknown environments.

Previous work

Critter and the Pack Rats

The earliest group robot activities at MIL centered around simple “puck-pushers.” Reid Harrison, a former UF undergraduate engineering student, and myself developed a group of simple robots capable of detecting and pushing a puck toward a light and following a black line on the ground. The robots could also detect a potential collision through the use of whiskers. The emergent behavior of this group was the collection of randomly placed pucks at a light source. These robots contained very simple programs (256 bytes on one robot) which used the central idea of value differentiation in the CdS cells. In one part of the program, the robot would look at the nose CdS cells and try to center the robot over a black line on the floor. This simple behavior allowed the robot to follow a black line on a white background. When the robot picked up a puck (i.e., started pushing it) a switch closed forcing a change from nose behavior (line following) to eye behavior (light following) that used a different set of CdS cells. The emergent behavior involved the robot following black lines until it hit a puck. Then, the robot would take the puck to a light source on a wall at robot “eye” level. Upon hitting the wall, the robot would deposit the puck. The atomic behaviors are “detect bump/puck” and “follow light/dark” (note that these are really just 2 behaviors with variable switches for “bump/puck” “light/dark”). With those two behaviors, the robot can achieve puck accumulation, line following, collision detection, and light following. In a group, the emergent behavior is a much faster puck accumulation. Without changing code, the robots automatically adjust to having more robots working with them. This experiment gave us the first indication that, if emergent behaviors were allowed to work, a group could self-organize with no global knowledge and (more importantly) no communication between the robots. The absence of communication being programmed into the robots is an important movement away from some other experiments in swarm robots where robots communicate expressly to each other. In the puck-pusher experiments, all robot interactions emerged from individual responses to local perception and not on direct communication with other robots. I feel based on these and other experiments that the advantage of non-communication is understated in robotics. Robot tasks that do not require communication scale more easily and behave more robustly than tasks that require communication, the experiments with TJs in the 4th chapter show this.

Gator 2 Swarm

During the Gator 2 (G2) development, we made the first serious attempt at multiple robot interactions in MIL. Pervious work involved simulations of multiple robots and single robot experiments along with the puck pushers. The purpose of this work was to develop robotic platforms which could work together and manipulate objects in their environment. Our goal for these robots was to simulate a factory floor where "machines" worked with "materials" and robots transported "materials" to and from the "machines". The "machines" were simply docking stations with beacons and the "materials" were pucks. We learned from this experiment that the complexity of a small mobile robot with a 2 DOF manipulator and limited sensors on it was a limitation in both the number of robots that could be built and the algorithms that could be implemented on them. Later robot designs included printed circuit boards (instead of hand wired) and mechanics that were cut by a CNC machine. These developments enabled us to do experiments with large groups of easily produced and maintained robots. This new generation of robots subscribed to the mass production paradigm. We constructed them from identical and interchangeable parts, and little or no knowledge of the robot circuitry was required for assembly.

Hardware

TALRIK

A TALRIK robot, the name "Talrik" derived from the Swedish word "tallrik" meaning "plate", is a circular robot controlled by a 68HC11 and 64K of memory with 12 IR sensors ringing the robot along with a bumper. The design of these robots came from the Grazer construction. The Grazer robots were circular to prevent the robot from hanging up if it collided with an object. Another design borrowed from the Grazer and implemented on TALRIK robots is the use of modified digital 40KHz sensors for analog IR light detection, which gives an indication of the distance to an object. Ariel Bentillo in the Machine Intelligence Lab, first developed this technique of hacking digital sensors.

TJ

TJ, or "Talrik Junior" is a smaller, simpler robot based upon a single chip 68HC11E2. This robot has basic collision avoidance capability with IR sensors and a bumper. It uses modified aircraft servos as motors. Again, the robot is circular and constructed out of aircraft plywood cut on a CNC milling machine and glued together. These features make TJ mass produceable, an important consideration in swarm experiments. The robot is also robust and durable. Among TJ (mis)adventures of note: being played with by forty kindergartners, being dropped from 6+ feet, and being mauled (and drooled on) by a 105 pound rottweiler. Hardiness is important in experiments where 30 robots are running simultaneously and there is no place for tangled wires or shorted circuitry. As an elementary educational tool, this durability is also very important. Because of TJs small size (6.5 in. diameter and 3.5 in. tall), the rechargeable NiCd batteries will last for 1.5 hours and long experiments are possible.

TJ-PRO

TJ-PRO is mechanically the same as a TJ, but the processor on board has an additional 32K of memory. This memory allows the robot to be used in some of the more advanced educational experiments. The programming languages for TJ-PRO make teaching younger students science engineering and mathematics through autonomous mobile robots an easier task.

CHAPTER 3

LEARNING TO EAT

Abstract

This chapter describes the implementation of learning algorithms for increased recharging efficiency in long-term autonomous recharging robots. Our goal is to demonstrate the advantages of a robot platform that can operate autonomously and continuously for a week or more. The learning algorithm focuses on regulating the robots "eating habits," defined as the robot's ability to regulate charging in order to maximize the moving time and minimize the charging time. The final analysis of the data obtained during our experiments indicates that the robot learned to stay in the most efficient region of the batteries' operation.

Introduction

This chapter documents our experiments with the Grazer robots. The goal of this research was to develop a robust robotic platform capable of surviving for an indefinite period of time without human intervention. I implemented learning algorithms in the agents to facilitate the longevity of the robots and to imbue them with a primitive instinct for survival. I chose implementation of "learning-to-recharge" partly because it is an inherently long duration process. Experimental run times greater than 24 hours are essential in order to obtain meaningful data with the Grazer robots. I also sought to demonstrate the advantages of using long-lifespan robots for the evaluation of learning algorithms. The Grazer robots provide an effective testbed for learning for a number of reasons:

1. Researchers can carry out long term experiments, allowing them to run numbers of trials that were once only possible with simulation.

2. Agents can compete for limited resources in a realistic manner since survival of the fittest is naturally occurring with these agents. Those robots which do not compete successfully die off.

3. Due to the continuous nature of the experiments, analysis techniques begin to resemble those used in ethology and lead to direct comparisons of synthetic ecosystems with real ecosystems.

4. Researchers can evaluate the fault tolerance of algorithms in ways previously impossible. An algorithm that can reliably and continuously run on a real agent for a week yields a greater level of confidence in its performance than an algorithm that runs in a series of two-hour segments.

5. The physical support system for embodied agents provide "natural motivation" not easily or realistically provided in simulation studies.

6. Implementations are more easily scaled to real world applications. The engineering problems associated with long-lifespan robots allow for a closer coupling between lab work and commercial applications.

Background: Ancestry

The development of long-lived, self-rechargeable robots has not received as much attention as it deserves. Feeding robots are by no means a new idea. During the 1950s, a vacuum tube robot "turtle" built by W. Grey Walter, could recharge itself (Walter, 1950). Recharging robots built by Luc Steels (1993, 1994) have demonstrated the potential of having learning robots cope with internal and external energy management. Learning mobile robots that cannot recharge are limited by their short run times. In such cases learning must be scaled down to fit into the lifetime of the battery pack. In her Ph.D. dissertation, Maja Mataric (1994) noted the limitations of short run time robots. These problems become obvious with robot swarms because of the large number of agents that require constant human attention to keep them running. Mataric's work with learning in groups of real robots has been an inspiration to our work.

Learning Algorithms

This research does not advance the theory of learning algorithms. Rather, this work provides a platform for evaluating learning algorithms in extended experiments. The various approaches to machine learning (neural networks, R-learning, Q-learning, etc.) are beyond the scope of this paper. Analyses of various methods employed in machine learning has been well covered in the literature (Mahadevan, 1994) (Mahadevan and Connell, 1991) (Jaakkola et.al., 1993) (Hexmoor and Nute, 1992) (Watkins, 1989) (Kaelbling et al., 1995).

For several reasons, the problem of recharging is appropriate for learning. First, since the number of variables defining a state is limited, the complexity of the learning is tractable. Also, the reinforcement evaluation is clearly defined, namely "minimize charging time with respect to running time." Given these criteria, mapping between the current state and the next state (i.e., to look for the charger or not) is relatively simple. Correspondingly, the evaluation of rewards is also less complex. The agent is rewarded if the action increased the average time running as compared to the average time charging.

Experimental setup and learning algorithms: Birth

The Grazer robots currently consist of four robots, three of which are capable of recharging. The other is used for sensor development. Because of the long duration of the experiments, multiple robots enable us to improve code and hardware without disturbing a robot that is in the middle of a multiple-day run. The robot circuitry consists of a Motorola MC68HC11 EVBU and an in-house expansion board that provides 32K memory and handles motor control and sensors. Grazer realizes collision avoidance by means of modulating an IR emitter and detecting the amount of 40KHz IR energy reflected from obstacles and turning away from them..

Grazer possesses a circular platform mounted on two independent drive wheels with a caster (Figure 1). This wheel arrangement affords two advantages while the robot navigates in an environment. The robot can rotate about its center, virtually eliminating the possibility of the robot getting trapped. Further, the circular design prevents the robot from catching onto an obstacle in the event that its collision avoidance sensors fail to detect an obstacle. These are important qualities for a robot which spends days without human intervention. Figure 1 also provides a view of the floor charge plates and the recharger box.

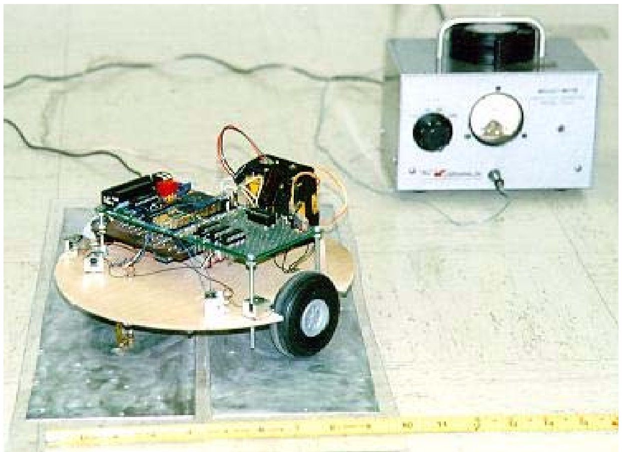

Figure 1: Grazer and its charger.

Charger

Two floor plates provide raw power to Grazer. The plates supply 28 VAC and a power resistor limits the current. The use of AC allows Grazer to approach the charger from any direction to obtain energy. No markers tell the robot where the charging station is located. The robot recognizes this "feeding ground" only when it accidentally rolls across it. We liken this behavior to a grazing animal ranging the grasslands, hence the name for these robots.

Environment

The world of Grazer consists of an irregular area of approximately 200 square feet of floor space interspersed with obstacles. The charging plates sit in no particular area and we can move them at will. Because this is also the main testing area at the MIL, Grazer must contend with daytime visitors, various robots, such as hexapods, and wheeled robots of many varieties, as well as human intruders wandering across the experimental area. At night the lab belongs to Grazer, with the possible exception of an occasional security person or janitor wandering through the lab.

Data Collection

A robot engaged in large unobserved quantities of running time must keep data on its performance and learning and report this data back to a computer for analysis. To keep the circuitry simple and affordable, we did not use a wireless link. Instead, the robot keeps several days worth of data in its memory. When it detects that a computer has been connected to its serial port, it sends the data to the computer. After data collection the robot returns to the experimental area to continue the experiment.

Experimental Results: A day in the life of a robot

Learning Goal

The goal of the learning task for the Grazer robot is to adjust the hunger threshold in such a way as to maximize the effeciency, defined by the ratio of running time to charging time. The only factor being adjusted by learning in these experiments is the hunger threshold. The hunger threshold is the battery voltage at which point the robot will respond to the charging plates by stopping and charging. The performance index used in these experiments is simply the average ratio of time spent moving and time spent charging. When quoting numbers in the data, the hunger threshold is an arbitrary number N proportional to the battery voltage. N = 150 indicates a fully charged battery pack and N=115 represents a discharged battery pack.

Fixed Ratio Experiments

To have a control for our experiments, we first ran them without learning, the robot recharges itself according to a fixed hunger threshold. This manually set threshold determined when the robot recharged. A threshold of 150 meant that the robot would charge every time it made a successful connection across the recharging plates, presumably the most inefficient charging scheme. A threshold of 120 meant that the robot waited until its batteries discharged more than halfway before recharging. The performance factor ρ, equals the average run time over charge time. The graphs plot to yield the run-time to total time ratio. Observe that if then $\varepsilon = 1/3$

$$\rho_{robots}[n] = \frac{\sum_{i=0}^{15} (\text{runtime}[n - i])}{\sum_{j=0}^{15} (\text{chargetime}[n - j])} \qquad \rho_{graphs}[n] = \frac{\sum_{i=0}^{15} (\text{runtime}[n - i])}{\sum_{j=0}^{15} (\text{chargetime}[n - j]+ \text{runtime}[n - i])}$$

This average over a window of data creates some anomalies. For instance, if the robot does not charge in 16 samples, the data can become skewed. Another side effect of this average is that it requires a minimum of 16 samples before the data becomes valid. Even after 16 samples, several more are required for the data to stabilize. This stabilization can be seen in the first hours of Figure 2.

Figure 2 shows the data for a run of 2.7 days with a constant hunger threshold of 150. The most obvious characteristic of the graph is the chaotic periods. Oxide and dust on the charging plates caused these erratic periods preventing the robot from connecting properly with the charger. With this information, we can see that the charger pads were cleaned at about 1.0 day and 2.0 days. Ignoring the oxidized pad regions, the average ρ is approximately 0.35. This value implies that the robot was moving for less than half the time that it was charging. This value of ρ represents the robots' worst performance possible short of death. With the data in Figure 2 as our baseline, we can explore the different learning algorithms.

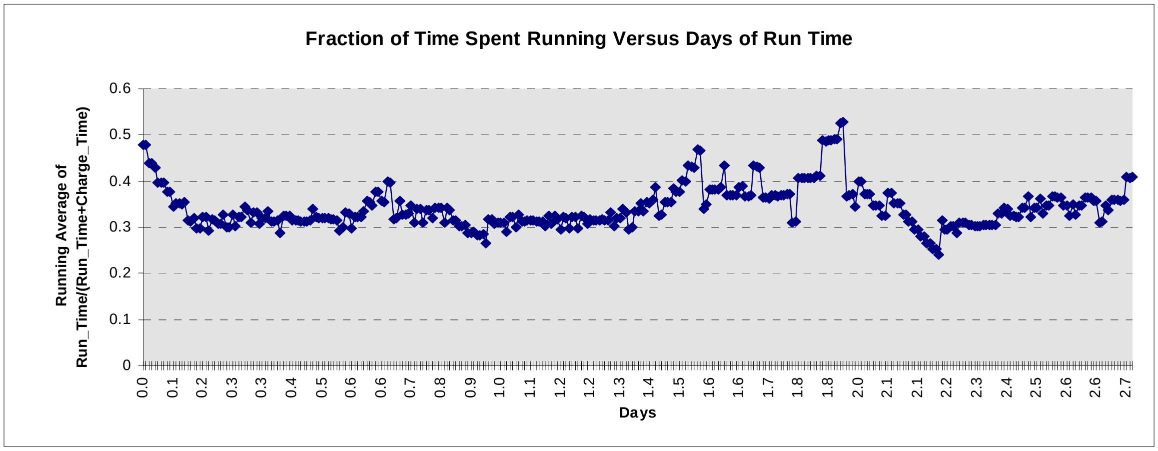

Figure 2: Performance of robot with fixed hunger threshold.

Self-calibration

Before implementing learning algorithms, we used a simple self-calibration technique. We use the term "self-calibration" to differentiate this technique from true learning. The difference between self-calibration and learning is that the tendency of the system is presupposed in the algorithm of self-calibration. Working from the experiments with a fixed hunger threshold, we infer two characteristics of the system. First, for hunger thresholds in the range from 150 to 120, a smaller value of hunger threshold will mean a better ρ. Any value above 150 will have the same effect as 150 on the system; therefore, the range from 255 (the limit of the sensors) to 150 can be discounted. The second inference is that at too low a value of hunger (approximately 110), the robot will starve to death before reaching the charger. The self-calibration technique assumes that there is some maximum between the points 150 and 110 that the robot can find.

The algorithm used to determine the hunger threshold is as follows:

for: $\rho[n] > \rho[n+1]$; hunger = hunger -1   for: $\rho[n] < \rho[n+1]$; hunger = hunger + 1

Based on this algorithm, the robot will decrease its hunger threshold until it no longer increases ρ . At first glance, the above algorithm appears counter-intuitive: the hunger threshold decreases when performance is deteriorating. However, the robot attempted to increase run time through this algorithm. We initially expected the robot to rapidly kill itself using this algorithm, but the experiment proved us wrong. By pushing ahead with the hunger threshold, Figure 4 illustrates that the robot survived, but only discovers a marginal improvement in performance from the fixed ratio experiment. Performance steadily increased over the duration of the experiment until the robot was spending nearly twice as much time running as charging.

In Figure 3, the threshold steadily marches downward. Unfortunately, the 2.6 day run was not long enough to see if this algorithm would stabilize at a lower threshold. The extrapolated time to stabilization is roughly two weeks for this algorithm. Recall that these are real robots being run for days at a time. Our current record stands at over one week.

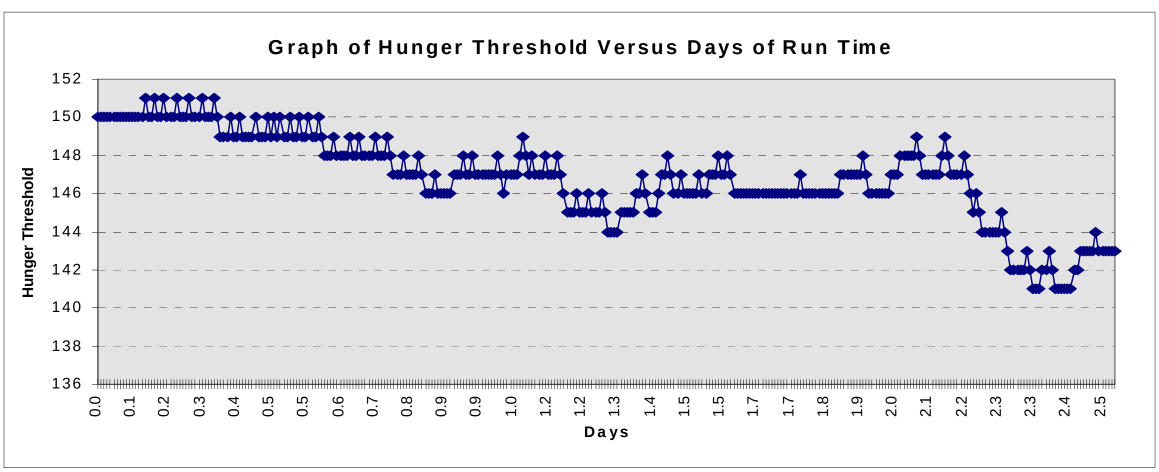

Figure 3: The change in the hunger threshold during the self-calibration run.

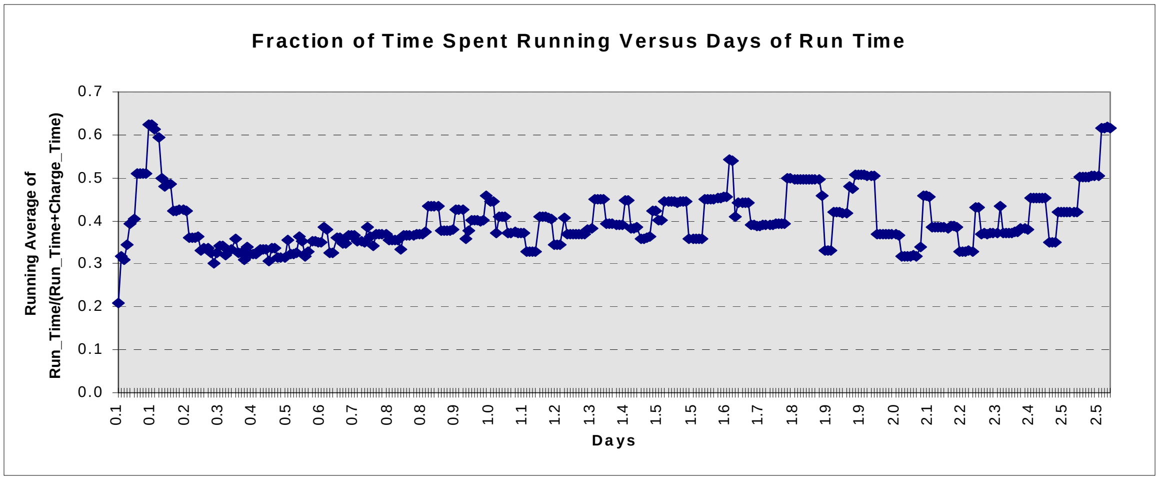

Figure 4: Performance during the self-calibration run.

Reinforcement learning

The general principle of adjusting the hunger threshold based on past performance improved the run time to charge time behavior. Using a simplified RL algorithm, the robot figured out for itself how to adjust its own hunger threshold. Reinforcement only applied to the direction in which the hunger threshold was moved.

The steps involved in the learning process were:

1. Choose a random direction[2] (increase or decrease by some fixed number of units) to move the hunger threshold.

2. Check if this direction improves ρ.

3. If ρ improves, then continue in that direction.

4. If ρ degrades, then reverse the direction.

5. Continue in the current direction until ρ ceases to change, then go back to step 1.

This algorithm implements learning in a simple way by defining states S as the ρ and the action A as the movement of the hunger threshold. In terms of this notation the state space is very simple and, thus, the reinforcement calculations are tractable. Figures 5 and 6 reveal the performance of the learning algorithm over a four day run. In Figure 5 the hunger threshold declines sharply over the first two days. We attribute this drop to the

[2] The random direction is supplied by the least significant bit of the system clock.

charger plates getting dirty. Our rationale for this hypothesis is that we cleaned the plates at about the 1.8 day region, which resulted in the learning algorithm increasing the threshold (as seen in the 1.9 to 2.8 day region). The algorithm ultimately converged at around a 30% run time, as seen in Figure 6. Because of this relatively low performance level (as compared to the fixed ratio experiments), we cannot conclude that the algorithm truly learned to increase its performance.

In this experiment, the learning algorithm simply responded to a change in its environment (i.e., the increasing difficulty of charging due to dirt on the charging plates). I feel that the failure of this algorithm resides in both the statistical basis of the performance and the physical nature of the charging system. To fully explain this data, I must execute many more experiments. The focus of future work will center on the robots' interpretation of its own performance. It is the robots' misinterpretation of its long-term performance that most likely caused the failure of this algorithm to produce the desired performance.

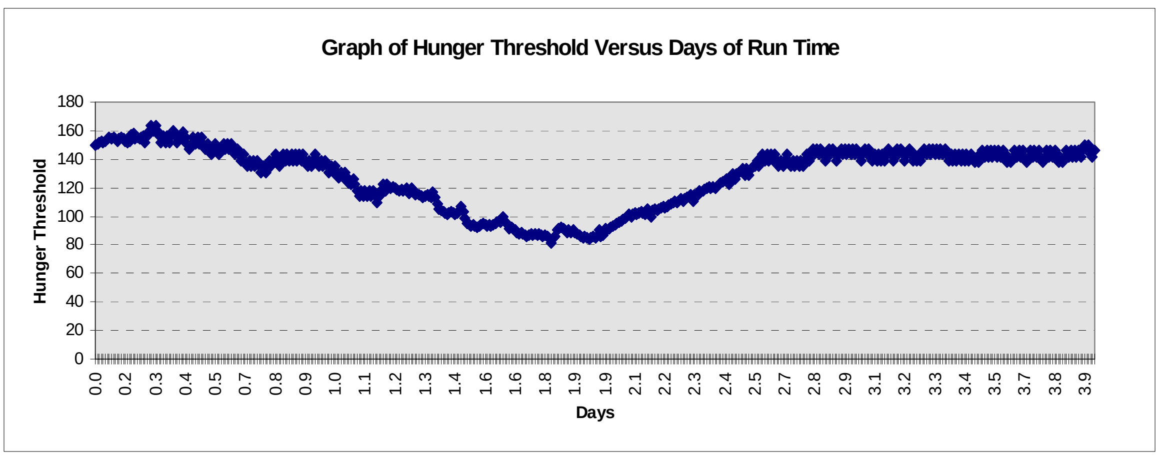

Figure 5: Graph of hunger threshold during RL run.

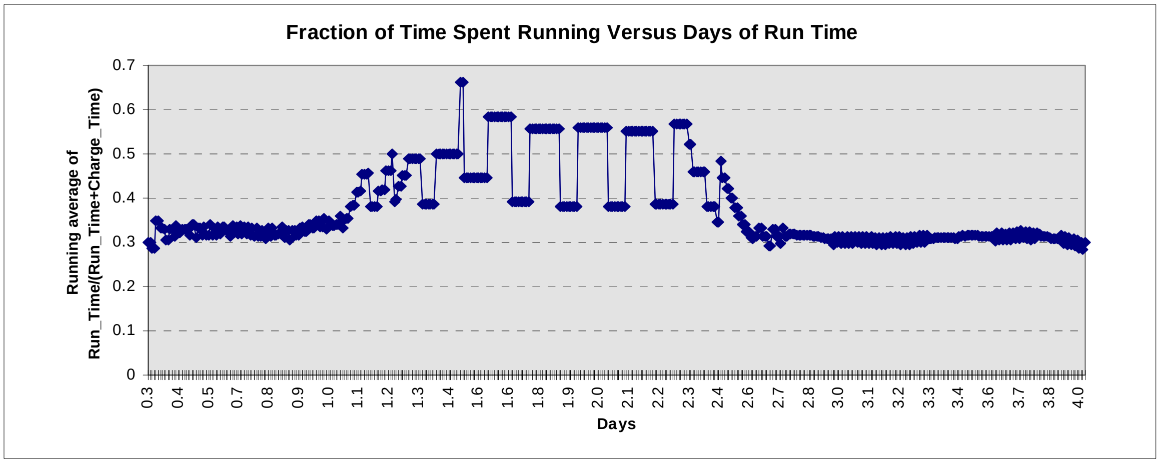

Figure 6: Graph of performance during RL run.

Problems and case studies: The will to live"

Robot manslaughter

One case study in the behavior of the Grazer robots that is particularly interesting is a murder that occurred during a run of two robots. As described earlier, the environment has only one charger. That charger is big enough for two robots only if the robots are perfectly positioned, which is unlikely to happen. The two Grazer robots are identical, except in their motors. Because of motor burnout problems encountered with the first Grazer robot, the second Grazer had higher torque motors. This fact, coupled with the difficulty in one robot sensing another from behind, led to this robotic tragedy. When we introduced the two robots to the test environment with only their obstacle avoidance and charging behaviors active, a new behavior emerged. When the stronger robot would approach the stationary weaker robot while the weaker one was charging, it would push the weaker one off the charger. When the opposite conditions occurred, the weaker robot was unable to push the stronger one off the charger. It would spin its wheels until its sensors finally saw the stronger robot as an obstacle, at which point it would give up. Within several hours, the weaker robot was dead and the stronger robot ran for another day.

Luc Steels (1993) defines emergent behavior as follows:

> A behavior is emergent if it can only be defined using descriptive categories which are not necessary to describe the behavior of the constituent components. An emergent behavior leads to emergent functionality if the behavior contributes to the system's self preservation and if the system can build further upon it.

Our example of robot "manslaughter" (I probably should not impart too much self awareness unto our agents to consider it first degree murder) seems to satisfy Steels' definition, especially since the stronger robot enhanced its own survivability by killing off the competitor for the limited resources of the charger pad. As an interesting aside, this murderous behavior was first noted by W. Grey Walter in 1950 when his "turtles" would compete for the charger (Walter, 1950).

Old Age

Numerous problems plagued the early Grazer robots. Serious reliability issues arise in getting a small autonomous robot to survive for 24 hours unattended, let alone for one week. Motor burnout came first. The inexpensive motors used on the first Grazer robots contained metal brushes which, after a couple days of use, burned out, and destroyed the robots' mobility. Installation of carbon brush motors solved this problem. Overheating of the motor driver chip did not show up until the robots could run for several days. We easily fixed the overheating problem with a heatsink. Our operation of the robots for a week or more required periodic cleaning of the charger contacts.

Conclusions

In the course of this research, we made the following significant achievements:

1. We constructed an inexpensive (<$100), small platform which ran autonomously, continuously and without human intervention for periods of time exceeding one week.

2. During the course of these experiments, we did not see noticeable memory effects in NiCads that would limit the lifetime of the robot. Memory in a NiCad may reduce the shallow-cycle lifetime of the batteries, necessitating a deep charge mechanism for reviving the full capacity of the cells.

3. We demonstrated the implementation of simple learning algorithms that require extended periods of time to run.

4. We also demonstrated data collection by the robot over extended run times, allowing the experimenter to analyze performance of the robot over days instead of experiments that are limited to hours and often minutes.

The data we obtained in these experiments was puzzling at first, and we felt that the learning algorithms were not performing as expected. The data generated by the robot's learning algorithm seemingly had failed to reach an optimum point. However, after discovering a graph from the Energizer battery data (Gates, 1989) that showed the relationship of charging efficiency to the degree of discharge of the cell, we began to make sense of the data. The bottom graph of Figure 7 shows that the charge efficiency is flat in the middle region 2 and drops off drastically for deep cycling area 1, and over charging areas 3 and 4. This explains why the learning algorithm never left region 2 because at any point in region 2 the robot was charging efficiently. When the robot ventured into the deep cycle region 1 it was forced back to area 2 (refer back to Figure 5). The robot seems to have learned what the battery manufactures knew all along about their batteries.

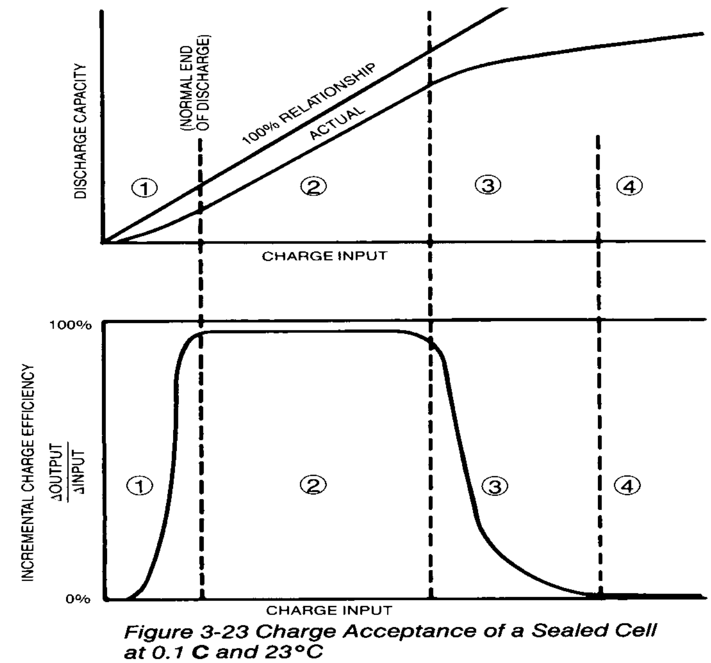

Figure 7: Efficiency of charging graphs. (Reprinted with permission, Everready Battery Systems)

CHAPTER 4

KINETICS OF ROBOTICS

Abstract

In this chapter, Drew Bagnell, Ivan Zapata and myself derive and apply a new technique useful in the analysis of swarm robotics. We derive useful swarm descriptions from the basic kinetic theory of gases. Several experiments verified our basic assumptions made in the theoretical derivation. We also performed experiments to test the methodology as a tool in evaluating robotic behavior algorithms.

Introduction

In the following experiments, we introduce a novel approach for analyzing robot group interactions. We based our approach on the kinetic derivation of ideal gas laws. The motivation for this approach came from the need for quantitative tools for the analysis of swarm robotics. Swarm robotic experiments currently involve chaotic experiments with no clear way to determine progress nor an indication of whether an algorithm is performing better or worse than expected. Without feedback as to the performance of the swarm, learning algorithms are difficult to train. The need for a more precise set of evaluation tools is a universal problem in swarm robotics. Mataric (1994) solved the analysis problem by tracking each robot as it moved. This method greatly increases the cost of the robot and also limits where the experiments can be conducted. Few people besides Mataric have worked with eight or more real robots operating simultaneously.

In simulations, data collection does not present a problem since the computer tracks the coordinates of each robot and their progress. The ease with which simulations track swarm robot movements has hidden an interesting avenue of research: how to measure swarm performance. Our approach was to use indirect measurements of swarm performance through the statistical interactions of the robots with each other and their environment. The statistical approach lends a clearer view of the complex interactions among swarm robots. Instead of tracking the path of every robot, we can minimize data collection and determine indices of performance using simple experiments. Another benefit of this work was the development of robot independent criteria. The robotics field has suffered because of the lack of repeatable experiments found in other areas of science. A researcher cannot easily reproduce the experiments of another researcher in robotics because of inconsistent data collection techniques and platform dependent issues. Our measurement techniques and derivations have no platform-dependent assumptions. Our goal is to make the measurement of swarm robot behavior an experimental science by describing experiments that can be reproduced by any robotics researcher working with a group of robots.

Justification of an Expression for Effusion

In an effort to model the simplest of swarm behaviors, Drew Bagnell and myself applied tools developed in statistical mechanics for the description of ideal gases (Atkins, 1994). Consider a swarm of robots N within a plane cell, with an opening in the boundary to allow robots to exit. Assume the swarm agents move in random directions at constant velocity. In the event of a collision with another robot or a wall, a robot executes a random turn and moves again with constant velocity.

Figure 8: Robots Colliding With Wall.

Let $\mathbf{Z_w}$ be the number of collisions of swarm robots with a wall per unit time per unit length, and $\mathbf{L_0}$, the size of the aperture through which the robot could escape. The rate of the swarm's egress from the confining cell equals the product of these quantities, under certain assumptions and approximations. Specifically, we assume that robots may not re-enter the cell (effusion into a vacuum), that the robots are small compared to the aperture in the cell, and that the cell wall has no thickness.

Given a wall of length **L,** and a swarm moving in random directions with a velocity ***c***, and x-component Vx, half of the robots in the area given by $L \times V_x \times \Delta t$ (see Figure 8) will collide with the wall. Contributions from robots originating outside the area cancel those leaving the area, and this quantity becomes negligible as L increases. Since there is no speed distribution (the whole swarm moves with constant velocity c), the average x-velocity for those robots heading toward the aperture is, .

Thus the number of collisions in time T equals where *N/A* equals the density of the swarm (robots per unit area). The expression for collisions per length per unit time is as follows:

Thus, from our model, the rate of change of number of robots confined to the cell is given as:

After separating variables and integrating, we arrive at a relation between the number of confined swarm robots and time,

where $\mathbf{N_0}$ is the starting number of robots in the swarm.

Justification for mean collision time between swarm robots

Considering the robot swarm activity as analogous to a perfect two-dimensional gas, we can derive an expression for the mean time between robot collisions. Consider a circularly symmetric agent TJ moving with relative speed, c_{rel} . Other swarm robots are considered frozen in the plane. A collision between robots is guaranteed for any other agent whose center lies within the rectangle, having length $c_{rel} \times \Delta t$ and height $2\times d$ (***d*** being the diameter of the robot), described by TJ's movement.

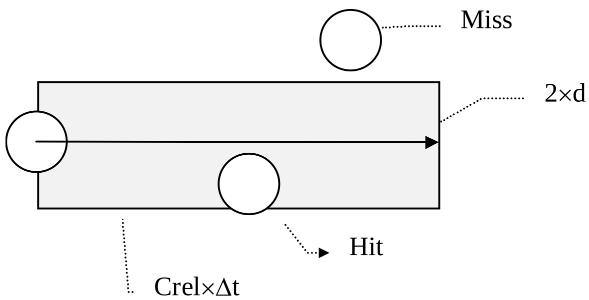

Figure 9: Diagram of Mean Free Path.

Under uniform distribution of velocity, direction c_{rel} equals . The expected number of collisions equals the area of the collision rectangle multiplied by the number density (N/A) of the robots. Dividing the expression by time leads to the following expression for collisions per unit time:

Taking the reciprocal of this expression gives an estimate of mean time between collisions. Multiplying the mean free time by an agent's speed ***c*** produces an expression for an agent's mean free path.

We based our approach to the analysis of swarm robot interactions on the kinetic derivation of the ideal gas laws. We expect that equations similar to those that govern the interaction of atomic particles will promote insight to the group behavior of robots. In molecular kinetics, the derivation of the gas laws is based on a 3-D case. In robotics, the derivation is clearly a 2-D case, and so a re-derivation of these key equations was necessary to make them applicable to swarm robots confined to a plane.

Robotic platform

The robotic platform used in our experiment is the Talrik Junior (TJ) robot produced by Mekatronix Inc. TJ (Figure) consists of a small 7 inch diameter platform with two drive servos for locomotion. The sensors include two 40KHz infrared emitters and detectors for obstacle avoidance and several bump sensors for collision detection. The on-board computer is the MC68HC811E2 8-bit microcontroller from Motorola with 2K of on-chip EEPROM and 256 bytes of RAM.

The power source for the robot is a six cell NiCad pack capable of recharging on the robot.

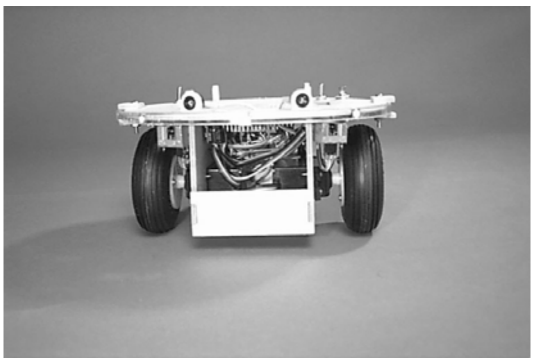

Figure 10: Front view of TJ robot.

Experimental setup

In the interest of facilitating other researchers to verify our results on their robot swarms and with simulation, this section will describe the experimental details. Only the scale of the experiments needs to change for independent verification on larger robotic platforms. Chemistry defines effusion as movement of molecules through an aperture. In our experiment, robots move in a 100 ft^2 area and then effuse through a 28-inch hole. In physical chemistry terms, the other side of the hole is a perfect vacuum where any robot passing through is shut off and removed from the experimental area until the next experiment. Figure 11 shows a picture of the environment.

Figure 11: Experimental setup.

Results

Collision frequency and collision cross section

The first step in comparing our theoretical kinetic model of robotics to the real robots interaction was to measure the collision frequency. From our theoretical calculations, we were able to determine a collision cross section for the TJs (14 inches) and, from this, a collision frequency dependent on the density of TJs in an area. In one experiment, we allowed four TJs to bump each other in a 25ft^2 area. We then counted TJ to TJ collisions on one robot over two minutes. In this experiment, we measured 0.172 collisions per second. Our calculated result for this density is 0.198 collisions per second (13% error). In the next experiment eight TJs were in a 50 ft^2 area, yielding the same density as before. This experiment ran for five minutes, yielding a measured collision rate of 0.143 collisions per second. The expected value again was 0.198 collisions per second (28% error). The final experiment was with eight TJs in a 100 ft^2 area. Which yielded an experimental collision frequency of 0.083 and a predicted value of 0.098 (16% error). These values verified what we expected: in a more diffuse case the robots act more like an ideal gas. The same holds for the kinetic gas laws. As a gas molecule becomes a significant percentage of the area (under extreme pressure), the experimental data begins to deviate significantly from ideal gas laws.

Effusion

Once the collision frequency experiments assured us that the equations did indeed reflect what the robots were doing, the next test was effusion. In the effusion experiments, TJs effuse through a hole in a wall and we record the time stamp of each TJ crossing the hole. We then compare the time stamps of TJs leaving the area to the predicted results.

IR off, bumper only

In the first experiment, the TJs executed collision detection. The TJ moved and hit an object, either another TJ or a wall. Once a TJ detected a collision with the front bumper, it then turned randomly and heads off in a new direction. We suspected that this experiment would most closely agree with our calculated results because of the similarity between the behavior of the robots and the behavior of molecules in a gas. Figure 12 plots the average results of five trials with the theoretical exponential result. This experiment coincided well with the result we expected. The discrete errors in the experiments are due to the small number of robots involved. Contrast eight robots to 10^{23} molecules in a gas!

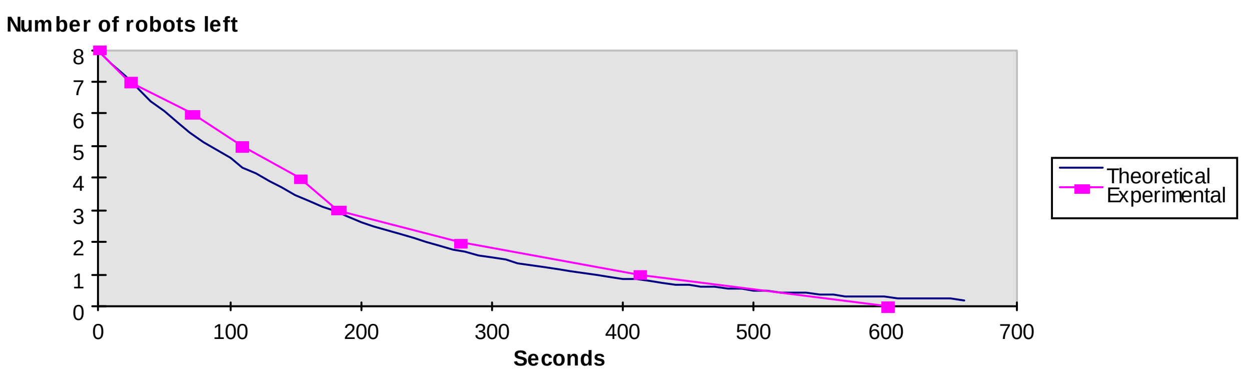

Figure 12: Average of Five Trials, Bumpers Only.

IR on, Braitenberg control

We based an IR collision avoidance algorithm on the work of Braitenberg (1984). The Braitenberg control is simple: one sensor crosses over and controls one motor in a purely reactive control. In this control scheme a wall on the left of the robot will be detected by the left IR sensor and causes the right motor to slow, thus turning the robot to the right (away from the wall). The converse is true for the right sensor. This control scheme contains some well-known flaws. The first flaw, which we call the “Braitenberg Trap,” occurs at a corner where the two sensors are balanced. The robot will oscillate for a prolonged period and may not be able to recover. Another problem with the Braintenberg algorithm is the tendency for robots to fall into set paths in an enclosed environment, typically circulating around without exploring the entire environment. We solved the first problem by using only obtuse angles in the environment to prevent oscillation. Initially, the complexity of multiple robot interactions took care of the second problem. However, as the number of robots diminished in the test area, the remaining robots tended to circulate and prolong the total effusion time. Figure shows the results of these experiments. The Braitenberg rotation traps led to the widely varying results. In two trials, the last robot failed to escape as it rotated about the arena in a fixed pattern. In Figure 13 shows an experiment with 26 robots demonstrating the difference between the high-pressure case and the low pressure case. The robots clearly effuse faster under higher pressure.

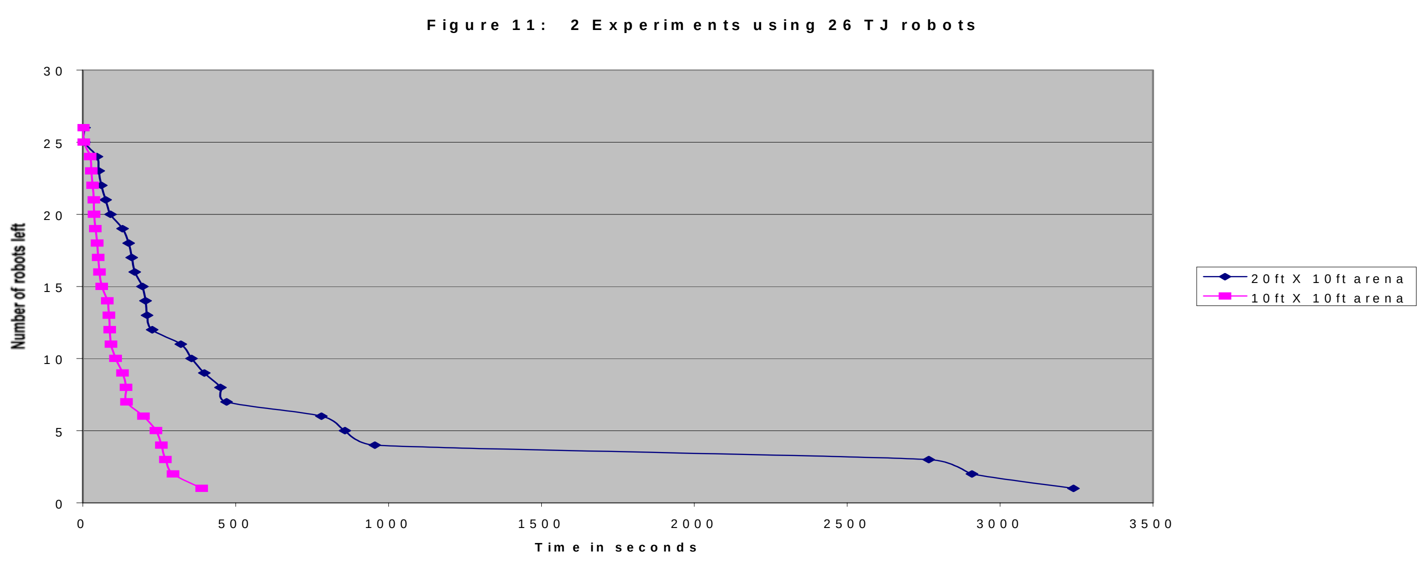

Figure 13: Differences in Effusion Rates for High and Low Pressure cases.

IR on, random turn

This algorithm is built around the robot seeing an obstacle with its IR and turning randomly. The test here was to evaluate the sensitivity of the theory to several conditions. The IR detectors increased the robots effective size, since the robots have the ability to detect an object farther away (approximately 10 inches). We did not factor this effect into the theoretical derivation since we assumed that the hole was big enough for the robot to pass through. Another consequence was that the effective area was reduced since the robots could detect the walls without hitting them. In Figure 14 the data for this experiment seemed much more consistent than the previous data. We believed that the faster average escape time was due to the increased "pressure" of the TJs, resulting from their detection of the walls from a distance. Thus, the area was effectively reduced.

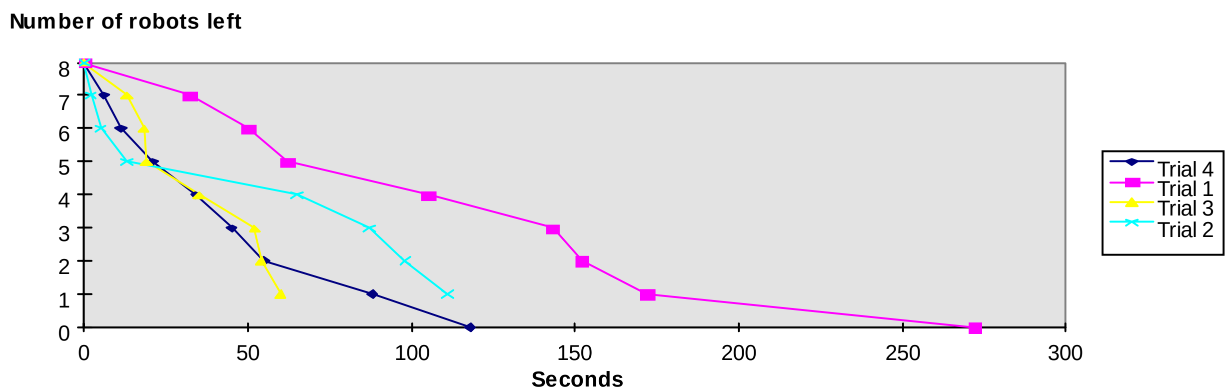

Figure 14: IR Avoidance on, Four Trials.

Following robots

In this experiment, the algorithm imparted a tendency for the robots to follow each other. When one robot leaves, others may follow. We expected the robot action to deviate significantly from the theoretical prediction in this case. In fact, the interaction of the robots caused the effusion rate to not be a simple exponential function. Figure 15 shows that the escape time was greatly reduced, almost by a third.

The one major deviation from this fast escape time occurred when the last remaining robot had no robots to follow out of the experimental area.

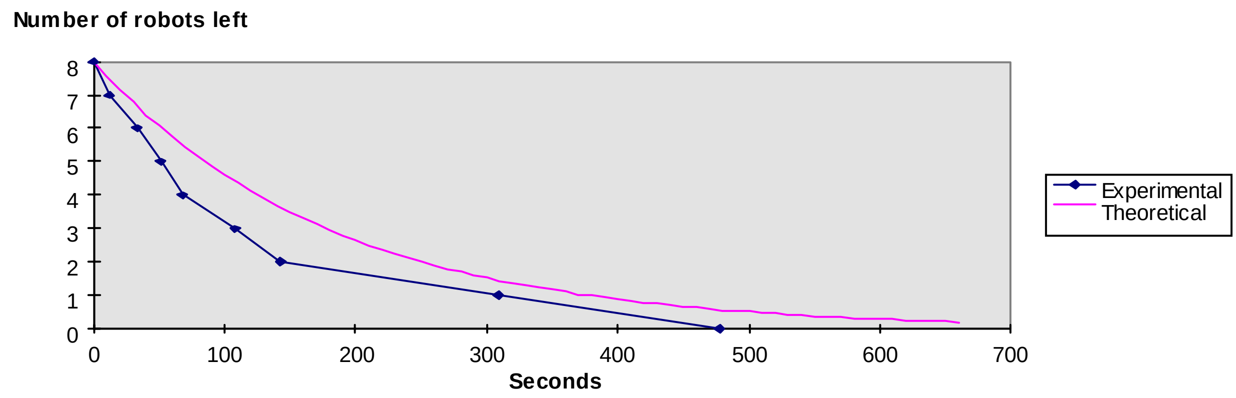

Figure 15: IR Avoidance on, Random Turn Average of Five Trials.

IR beacon

This experiment represents an extremely non-ideal gas case. We placed an IR beacon by the exit to attract the robots toward the opening. This experiment approximated particles under the influence of an external force, such as an electric field. The data in Figure 16 show the rapid exit rate at which these robots escape when the exit has an attractor.

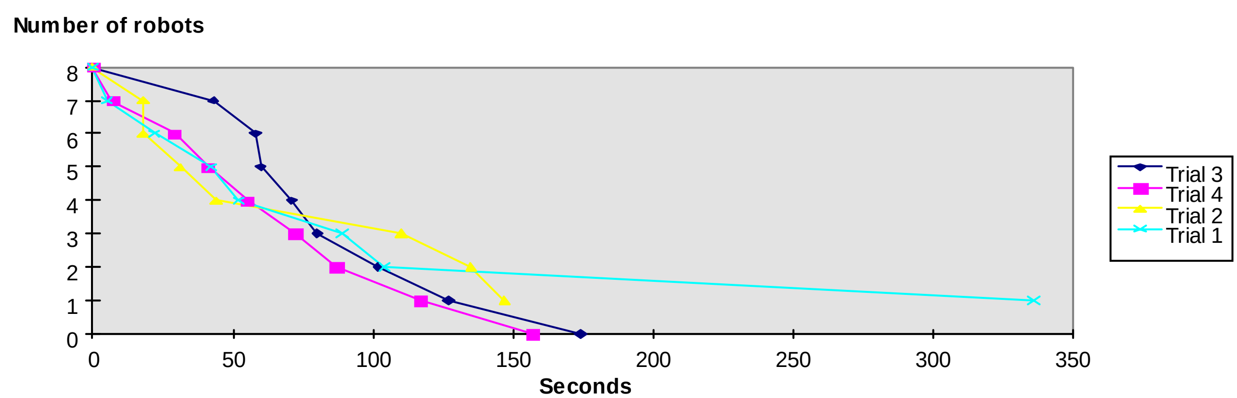

Figure 16: IR Following.

Simulations

To perform experiments with agents numbering in the hundreds, we had to turn to simulation. Simulation also allowed a fast way to explore different experiments and to verify the results in the robotic experiments. We performed these simulations in StarLogo (Resnick, 1997). In our simulations, we had to introduce a randomness component to make the simulations work. If we removed this randomness, the simulations degraded into stable oscillators and nothing was achieved. Real robots require no randomness because the environment and inaccurate sensor reading provide essential randomness. Our first simulated experiment dealt with the effusion of agents running simple collision avoidance. In Figure 17 runs 1 through 4 detail our baseline effusion experiments. Run 5 shows effusion with an aperture half the size, and run 6 shows effusion with the volume reduced by approximately ½ and the same aperture as run 5. Decreased volume increases the pressure in the arena, and thus increases the effusion rate.

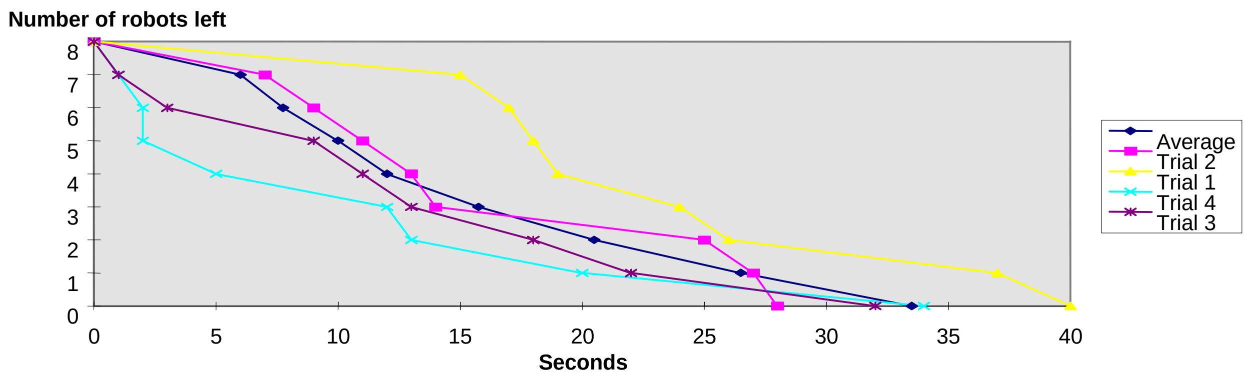

Figure 17: IR Beacon Four Trials with Average.

Conclusions

We illustrated that a useful technique for analyzing swarm robot interactions can be derived from kinetic gas laws. We applied this methodology to several experiments involving various algorithms in order to quantitatively evaluate those algorithms from the lowest baseline (robots moving randomly) to directed effusion (robots following a beacon to the exit). The quantitative characterization of a baseline in this metric also helps to evaluate the effectiveness of an algorithm in the accomplishment of a task (here finding an exit). In the past, most robot swarm experiments were highly subjective. With the development of a methodology and a theoretical model, we can begin to quantify algorithms and their ability to execute a task.

Future work

We expect this work to blossom into an exciting area of swarm robotics research. We have begun to develop the tools necessary to study synthetic ecosystems, learning algorithms and cooperating agents in a more scientific way. In the future, we plan to expand this theory to evaluate a variety of algorithms, to include experiments with a greater number of robots, and to add elements of biology, physics and chemistry.

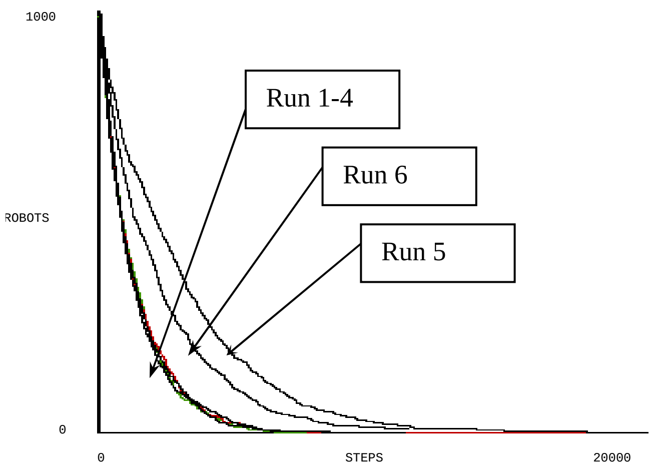

Figure 18: Effusion Simulation Data.

Figure 19: Our Complete Swarm of TJs.

CHAPTER 5

TJ CLUSTERS

Introduction

Along similar lines of research as the kinetics of robotics area, I then began to explore some other swarm behaviors rooted in physical laws. These behaviors are flocking, accretion, following and repulsion. The basis for these experiments derive from colloidal suspensions where small groups of particles aggregate together. Several reasons exist for looking at the behavior of atoms. They seem to be related to the formation of life (in chemistry of membranes and self-organizing molecules) and they have relevance to simple swarm behaviors in robots (like flocking).

TJ cluster experiments

In colloidal suspensions, the intermolecular (attractive) forces outweigh the dispersive forces which would try to separate the colloidal groups. However as the groups get bigger they do break apart. These basic rules also apply to properly programmed robots. The IR acts as the intermolecular attractant (acting over large distances as in colloidal molecules' attractive force) and the bumper models the repulsive force (when the robots get too close they move away). In the TJ program that mimics collusion, the robot will spin until it sees an increase in the amount of IR in a certain direction. It will then drive toward that IR. The IR attraction is done with a windowed average which allows the robot to constantly take the "relative" IR level. With an IR beacon, the robots will see IR values from very high (in a close group with other robots) to very low (in a disperse group). The windowed average allows the robot to deal with both situations. The robot will spin in place until it sees a spike in the IR values. It will then move toward other robots, creating a gradient of attractant IR. Since each robot emits IR, the greater number of robots attracted increases the gradient to attract robots. I verified this behavior in experiments with the robots. Even with initial orientations not conducive to forming a stable group (such as a long line), the robots still rapidly attracted to form a tight group.

The purpose of this experiment was to explore the relationship of flocking behaviors to colloidal suspensions in chemistry. My experiments showed that the robots would react to the gradient field emitted by their own IR. In my next experiment, I set out to see what would happen if there was also an external IR field applied to the environment. This experimental design mimics a flocking behavior, but it is also akin to the movement of ions in an electric field. The experimental design was to have a number of TJ robots start some distance from a beacon. Another robot emitting IR but not moving served as the beacon. I would turn on the robots and record the time for at least one of them to bump into the beacon. I recorded the data from these experiments in the following table (Table 1). The column labeled "initial heading" refers to the initial direction that the robots traveled: 180° designates away from the beacon and 0° designates directly toward the beacon robot. In the column titled "time to impact," the L's designate a lost group of robots. The cluster of robots wandered aimlessly until they encountered a wall and "stuck" to it (the wall was about 6 feet away in all directions).

Table 1: TJ clusters data.

Distance to Beacon (ft.)	Number of Robots	Initial Heading	Times to Impact (M:SS)
5	1	180°	0:32, 0:35, 0:30, 0:26, 0:28, 0:32, L, L
5	2	180°	0:44, 0:39, 0:53, 0:37, 0:54, 0:43, L
5	3	180°	1:07, 1:23, 0:27, 0:26, 0:23, 1:25, L, L,

			L, L, L, L, L, L, L
5	3	0°	0:54, 0:41, 0:43, 0:25, 0:54, 0:29, L, L, L, L, L
5	4	0°	2:36, 0:26, 0:43, 0:16, 0:45, 0:30, L, L, L
5	4	180°	0:50, 0:45, 0:26, 0:50, 1:15, 1:09, L, L
5	5	180°	L, L, L, L, L, L
5	5	0°	1:49, 0:42, 0:19, 0:15, 0:30, L, L, L, L, L, L, L, L, L, L, L
4	5	0°	0:21, 0:30, 0:17, 0:15
4	5	180°	0:19, 0:10, 0:34, 0:46, 0:42, 0:26, 0:11, 0:25
5	5	180°	:46, L, L, L, L, L

Because the robots both follow IR and emit IR they cluster as they drift towards the IR beacon. From this data, we can infer several things. The IR gradient increases in a cluster area due to the higher concentration of IR. There are weaknesses to this gradient approximation. One weakness is that the IR is emitted only from certain locations on the robot, the robot does not uniformly glow with IR. Another problem with this hypothesis is that, as a group of robots increases in density, some robots are actually shielded by other robots. The shielded robots do not contribute as much IR to the field. Even with these weaknesses, we would still expect the robots to follow the IR beacon until their own IR gradient field became high enough to offset the beacon and produce a stable cluster (due to more robots in the cluster). The data above reflects this progression. The number of trials that result in “lost” robot clusters (or clusters that never find the beacon but instead find a wall to attach to) increases as the clusters get bigger. The time to impact with the beacon is rather inconclusive because the distance is so short that the deviation in the times is greater than the progression in the data. The more robots in a cluster the more stable the cluster becomes. To verify this relationship after I reached the 5 feet / 5 robots stable cluster, I varied the initial orientation and distance to the beacon. Initial orientation had some effect on the experiment, but not nearly as great as the distance to the beacon effect. As one would expect from any line of sight radiation, the IR intensity gradient falls off as approximately $1/r^2$. This relation is an approximation because the LEDs are focused light instead of a true point source. Therefore, the change in distance from 5 feet to 4 feet (20%) would increase the IR intensity of the beacon by approximately 45% using the $1/r^2$ relationship. This change is enough to cause the cluster to always reach the beacon.

A phenomenon less obvious and more difficult to explain is the greater stability of odd (or prime) numbered clusters. In the data above, a cluster of three robots is more stable and less attracted by the beacon than a group of four robots. Unfortunately, there were not enough robots available to continue this trend at the time of the experiment. What about the odd-numbered groups makes them more stable than the even numbered ones? The packing of the robots may provide a partial explanation. Groups of three and five form triangle-like clusters, due to their common size and circular shape. Groups of two and four seem to “exclude” a robot, leaving it away from the central core and therefore more likely to see (and be attracted to) the beacon. The similarities between chemical systems and robotic systems appear to be more than incidental. The robots running this clustering program behave like colloidal suspensions. They begin to show signs of stability in certain configurations that mimic membranes and perhaps the self-organization of chemicals into life. The idea behind these experiments is to show the power of simple behaviors based on physical laws (attraction, repulsion, diffusion etc.). Emergent behaviors arising from robot groups that run these simple programs begin to approach social interactions in animals. The clustering resembles a herd or flock, and under the influence of a beacon, the robots resemble ants following pheromones. In simulation, “ants” in StarLogo will follow their own pheromones much like these robots being attracted to their own IR. The ant’s follow-pheromone behavior produces the emergent behavior spin-in-place when a large quanity of pheromone has been placed in an area. The emergent behavior of the follow-IR behavior in the robots is to stick to objects in a uniform manner. Robots attach to the wall, followed by others in the group. Once a robot attaches to the wall, it will tend to stay there, while robots in the cluster constantly move around until they are against the wall. The reason that the robot-on-wall behavior is a stable configuration is due to the wall reflecting back IR to the sensors (both are stationary on the robot platform). The IR coming from another robot moves separately from the receivers on the robot, making it unlikely for the robot to remain in one place. This dispersion along the surface of a wall is similar to capillary action in liquids which arises, from surface tension in the liquid. Since surface tension is due to molecules of a liquid being attracted to each other and at an interface, the attraction occurs only along the two dimensions of the surface and applies itself at the edges. The same effect is happening in the robots when the two-dimensional forces are converted to one-dimensional forces along the wall. The robots attraction to the walls results in these “forces”, (actually the “force” here is emergent the only action is between robots there is no explicit action by the wall) just as in water the hydrophlic forces that bind water molecules together. In water, hydrophila is due to hydrogen bonding, the electrostatic forces that the small hydrogen atom induces between molecules (Atkins, 1994).

My experiments have demonstrated part of the connection between physical laws and group emergent behavior. These experiments have also demonstrated the ability of robots to serve as instruments in these experiments. Robotic platforms allow a special level of control not easily obtainable in either animal or chemical experiments. Using robots in these experiments enables the researcher to have control over all aspects of the experiment, from the behavior of an individual to the construction of the environment. The ability to control these aspects does not guarantee that the experiment will be predictable or successful. To the contrary, the nature of emergent behavior almost precludes predictability. However this level of control does provide the experimenter with an understanding that certain atomic behaviors will lead to certain emergent behaviors. In animals, the inability of researchers to turn off behaviors (except surgically) greatly complicates research. This inability makes it difficult to determine, with any certainty, that a certain individual behavior will lead to a specific emergent group behavior.

CHAPTER 6

TJ AS EDUCATOR

Introduction

In the previous experiments, I tried to show the power of swarm robotics and emergent behavior for engineering research and science. The areas of robotics, swarm behavior and synthetic biology hold great promise not only in the areas of robotics and electrical engineering but also for engineering in general, biology, economics, ecology, environmental science and others. How can we expand this powerful new way of thinking to educate non-robotics people in the lessons to be learned from robots. This new perspective views the world as an ecology of machines and nature. Just as we have incorporated computers into education to help prepare students for a highly computerized world, we must now incorporate robots into education to prepare students for a highly robotic world, and to broaden their perspective of engineering the synthetic and natural world.

The new paradigm

I differentiate the previous paradigm from the new paradigm by using the terms convergent engineering and divergent engineering. I classify traditional engineering as divergent. In this style of engineering, mammoth creations are created by large teams sometimes in different areas of the world, working on small pieces of the total problem. In software design, this object-oriented approach to engineering focuses on small parts of the problem which are each individually tested and eventually brought together into the whole. The problem with this approach is testability. Each part is tested, but the whole is impossible to test fully. Engineers cannot test all states and all possible emergent states created through unforeseen interactions. As demonstrated in the design of a Boeing 747, the situation which recently brought down Flight 800 never appeared in testing the aircraft and never showed up in the millions of miles the aircraft flew. While it is already impossible to test all states of a device, with 2^{100} states even testing the important ones is challenging as technology increases in complexity. In divergent engineering unintended interactions between the parts and the environment (or agents and the environment in ALIFE terms) result in catastrophic failure. One small part fails and the whole plane falls apart. The way engineers have compensated for this problem in the past has been to over engineer, to design parts to be tougher than they need to be just in case something goes wrong. The result has been inefficient designs that are wasteful, expensive and *still* susceptible to catastrophic failure. In convergent engineering, the interactions between parts (or agents) is understood. Instead of fighting these interactions, the design takes advantage of them to make a more robust product. Convergent engineering is more biological in nature. Failures in parts may make the device slower or less efficient, but it will not fail simply. An engineered result is obtained by the parts converging in cooperation on the goal. In divergent engineering, the parts try to keep each other or an external force from tearing them apart.

TJs in the classroom

The advantages to this new paradigm are clear: increases in safely, reliability, reduction in cost and the ability to create more complex technologies than possible today. What we need now to incorporate this new perspective into the education of students. Although the experiments that time allowed me to perform in terms of education were limited, they did show the promise of using robots in education. In four trips to middle schools, our lab introduced sixth, seventh and eighth graders to robotics through TJ. In one class, we introduced students to the concepts of experimental biology through their work with the robots. They were to examine the robots as if they were some new animal. The first experiments involved TJs that were pre-programmed with a collision avoidance program. This program followed a very simple algorithm: if the robot saw something on the right, it would turn left and vice versa. This algorithm leads to the problem of oscillation at corners; the robot cannot decide which way to go. Students experimented with the robots in an attempt to understand how the robots perceive and act on the world around them. Through these experiments, the students learned about the principles of autonomous robots and ethology along with an understanding of what science means to them.

Shortages in computer programmers and other high tech skill areas have been reported (The Institute, February, 1998). Another statistic is that many students become discouraged and lose interest in engineering, science and mathematics in the middle school years. TJ answers the question of how to get (and keep) middle schoolers interested in math, science and computer programming. During these same trips and in other demonstrations since then, we also taught some of the students to program the robots through a Logo-like language developed by Drew Bagnell. Several interesting observations came out of these sessions. The potential for math education was clearly demonstrated by the sessions with the students. We showed students the basic movement commands (forward, left, right, stop) and encouraged them to explore the possibilities. One of the first programs the students wrote was to make the robot go forward, turn around and come back to where it started. A common mistake the students made was telling the robot to go forward and turn 360° instead of 180°. Comprehension was immediate. As the students watched the robot go forward, spin, and continue going forward, they truly grasped the physical meaning of angles. Many students shy away from mathematics because they fail to see the connection between math and the real world. Programming robots can help students make that connection. As side benefits, programming robots teaches students the importance of good spelling and communication. If a robot command is misspelled, the robot will not understand it. The robot will also only do what the student writes in the program. Through this, students learn the importance of precise communication skills. We also teach basic computer skills through robot programming. The actions of a robot easily demonstrates program flow along with procedural and recursive concepts. The robot will do this, then that etc. Although an apparently simple concept, it is one foreign to most middle school students. We also introduced the concepts of simple loops to the students, leading ultimately to more complex algorithm and sensor based projects.

The power of robots as educators could be extraordinary. As well as motivating students in math, science and computer programming, robots also introduce students to emerging technologies and new approaches to engineering, including convergent engineering. Another area for the use of robot-as-educator is in behavioral biology, where robots can provide a unique experimental perspective into the root of animal behavior. In a more traditional domain of robots, robot manufacturing can be modeled in the classroom. Our experience with students and robots demonstrates robot oriented education as an effective motivating tool for science education.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

Conclusions

In this thesis I have attempted to link the fields of electrical engineering, ethology, education, physical chemistry and computer science with the common thread being robotics. Understanding how complex systems work together as in insect colonies, large scale engineering projects or molecular interactions has profound implications for the technology in the 21st century. The topics I have begun to investigate impact diverse areas. Understanding how simple (or atomic) behaviors of animals, molecules or man-made objects interact to create unexpected and complex emergent behavior is critical to the understanding of many fields. In biology, understanding emergent functionality will help to explain how insect societies assemble themselves. Also of great interest to biologists is the understanding of ecologies. Of special importance to human endeavors is the understanding of how much an ecosystem can be disturbed yet not collapse. These experiments in emergent behavior may lead to a greater understanding in synecology (The study of the ecological interrelationships among communities of organisms).

As I have shown in the robot experiments, an emergent behavior can be very stable under perturbations, as in the cluster experiments. Elements of emergent behavior share a common theoretical basis with chaos theory. The stability of emergent robot behavior represents a version of the strange attractor in behavior. At the other end of the stability spectrum, the system can be very sensitive to small perturbations. In the Grazer research, the addition of a slightly stronger robot repeatedly killed off the weaker robot. This sensitivity relates to another chaotic system idea, the butterfly effect, which is the idea that a small perturbation in the system can cause a large change in the end result. In human endeavors, catastrophic failure in large complex systems (such as airliners) is a significant risk, since all aspects of a system cannot be known or tested. Engineers should design the system with the understanding that failures or untested states should converge to successful and safe conclusions and not result in catastrophic failure. Application of emergent behavior concepts into design may result in more reliable devices. I introduced the ideas of convergent and divergent engineering when used to describe emergent behaviors. The idea of emergent behaviors is also highly applicable to computer science. With object oriented programming, one can assemble large systems. Even though the programmer tests each part or object, the emergent function of the atomic objects may cause the catastrophic failure of the program under untested use. Another area of interest is the study of economies, the human societal equivalent of ecosystems. Understanding what keeps economies healthy and growing is of interest to people and governments and is deeply related to emergent behavior and the science of ALIFE. Teaching students to think of convergent engineering in their studies will lead to better engineering in the future and the stimulation of interest in science and engineering from elementary school through college.

The experiments in this thesis lend greater understanding of the relationship between programmed behaviors and emergent behaviors that result from their interactions. This insight has the potential to explain many different questions in science. The first question that this research begins to answer (although very nascent in its development here) is how the formation of life from basic molecular interaction is possible. I have shown through these robot experiments that very simple interactions based on physical laws of molecular interaction could result in emergent behaviors that are similar to those seen in colonies of insects and other animals. Another question which these experiments shed light on was the nature of competition and fighting. The Grazer experiments show that simply the existence of a limited resource is enough for the emergence of a fighting or competing behavior to emerge. This experiment shows Darwinian processes at work at the lowest level and without the need for explicit predator/prey behaviors.

Future work

The research here has just scratched the surface of what can be accomplished with autonomous robots in direct applications, research and education spin-offs.

Implications for robotics

Direct application of this work in the areas of manufacturing, consumer robotics, space exploration and software agents will have profound implications. Current robots in manufacturing are simply programmed machine tools with little ability to sense their surroundings and adjust their program accordingly. The robots' programming limits what they can accomplish. Re-tooling the assembly line may require reprogramming the robots on the line. Flexible manufacturing practically *demands* autonomous robots. Autonomous robots respond to changes in the manufacturing line and reconfigure themselves. Groups of robots can solve difficult scheduling problems. Through swarm interactions leading to emergent behavior, robots could tackle the supply and demand scheduling of materials handling on factory floors without the complexities that normally accompany scheduling problems. A similar system exists when insect colonies feed themselves. The insect colony sends out gatherers for food without a central plan for how much food is needed. The insect colony self regulates supply and demand through individual feedback, the emergent behavior of which is a properly fed colony. There is some evidence (in the failure of planned economies in the world) that central planning becomes unobtainable as complexity reaches a certain level while independent agents can provide a solution. The failure of centrally planned economies provides an example of this problem. Although other factors contributed to their downfall, central planning proved to be an intractable scheduling problem for a national economy. Apparently, to control every aspect of an economy from a central vantage point is virtually impossible.

Consumer robotics, including robot vacuums, lawn mowers and robotic cars, could easily surpass the personal computer in economic scale. Personal robots could also take on an entertainment purpose as synthetic pets and for educational purposes. The behavior of robotic vacuums and lawnmowers may originate from the survivability of the Grazer robots.

Space robotics requires the use of autonomous robots. Because of light speed delays, remote control becomes impossible in the outer planets. The planets in our solar system are bathed in radiation and have toxic atmospheres. To send people to these planets, we must first build structures to protect them from the environment. These structures will be built by robots preceding the human colonists. The ultimate robotic exploration was proposed by Enrico Fermi wherein a single robotic ship with frozen embryos on board could populate every star system in the galaxy in 1 million years with current rocket technology. Upon reaching a new star system 50 or 100 years after leaving earth, the robotic ship would use local raw materials with its onboard robots to construct habitation for the frozen passengers. Once these habitats were completed, the passengers would be thawed, and raised to maturity by robots to reside in their newly constructed habitats. Its work finished the robot ship would then construct a copy of itself. The two ships would then head off to new star systems. Although this scenario may seem far-fetched, it represents the most practical and lowest cost way to colonize the galaxy.

With the explosion of the Internet, software agents (non-physical robots) routinely move around in cyberspace. Software agents are the closest currently existing entities which are truly independent synthetic organisms. Because software agents can exist in the controlled arena of a computer memory or network they do not encounter such mundane and taxing problems as falling down stairs. Although not the focus of this thesis, the power of these agents provide a view of what autonomous agents can do for us when they enter the physical world in large numbers.

Implications for biological research

Perhaps the field most affected by robotics from a theoretical basis is biological science. Advances in robot learning and the understanding of group and individual behavior has come through experiments with robots. The key advance for biologists is the concept of control. In animal experiments, scientists never exert total control over all variables. In robot implementations, researchers can turn off parts of the "brain" with absolute certainty. This new level of control over experiments allows researchers to pinpoint how behaviors form and the ingredients necessary for learning. Artificial life is giving us the tools to look at natural life in a whole new context. Through robotics and ALIFE, the biological sciences will finally answer some of the most difficult questions surrounding animal and human behavior and the evolution of life.

Implications for education

I believe that students and teachers will harness the power of robotics in education. Robotics will create and sustain interest in the sciences, engineering and mathematics leading to a better educated student body. This interest will have a profound impact on future technologies and the social order of our world. The insights and advances made in robotics will leave no area of our world untouched. New technologies and new ways of thinking will create friendlier machines that are more efficient and durable than their predecessors, leading to great advances in our society and the eventual colonization of space.

REFERENCES

Agah, A. & Bekey, G.A. (1994). A novel cognitive architecture for simulated robots in an artificial world. *Proceedings of the IEEE Robotics and Automation Conference, USA*, 2309-2314.

Anderson, T.L. & Donath, M. (1990). Animal behavior as a paradigm for developing robot autonomy. In P. Maes (Ed.), *Designing autonomous agents*. Cambridge, MA: MIT Press.

Atkins, P. W. (1994). *Physical Chemistry*. New York, NY: W.H. Freeman and Company.

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

Brooks, R.A. (1990). Elephants don't play chess. In P. Maes (Ed.), *Designing autonomous agents*. (pp. 3-15). Cambridge, MA: MIT Press.

Carlson, N.R. (1994). *Physiology of behavior* (5th ed.). Boston, MA: Allyn & Bacon.

Cabanac, M & Lafrance, L. (1991). Facial consummatory responses in rats support the ponderostat hypothesis. *Physiology and Behavior, 50*, 179-183.

Caselli, S., Doty, K.L., Harrison, R.R. & Zanichelli, F. (1994). Mobile robot navigation in an enclosed large-scale space. *Proceedings of IECON94 Conference, Bologna, Italy*.

Chen, Q. & Luh, J.Y.S. (1994). Distributed motion coordination of multiple robots. *Proceedings of Intelligent Robots and Systems Conference, Munich, Germany*, 1493-1500.

Curry, G.L. & Feldman, R.M. (1987). *Mathematical foundations of population dynamics*. College Station, TX: Texas A&M University Press.

Doty, K.L. & Harrison, R.R. (1993). Sweep strategies for a sensory-driven, behavior-based vacuum cleaning agent. *Proceedings of AAAI Fall Symposium, Raleigh, NC*.

Doty, K.L. & Seed, S.L. (1994). Autonomous agent map construction in unknown enclosed environments. *Proceedings of MLC-COLT Workshop on Robot Learning, New Brunswick, NJ*.

Doty, K.L. & Van Aken, R.E. (1993). Swarm robot materials handling paradigm for a manufacturing workcell. *Proceedings of IEEE International Conference on Robotics and Automation, Atlanta, GA*, 778-782.

Gates Energy Products, Inc. (1989). Sealed rechargable batteries: Application manual (1st ed.). Gainesville, FL: Author.

Goodenough, J.L. & McKinion, J.M. (Eds.) (1992). *Basics of insect modeling*. St. Joseph, MI: American Society of Agricultural Engineers.

Harinarayan, K.R. & Lumelsky, V.J. (1994). Sensor-based motion planning for multiple mobile robots in an uncertain environment. *Proceedings of Intelligent Robots and Systems Conference, Munich, Germany*. 1485-1492.

Harrelson, K. (1994). Medusa. Unpublished manuscript.

Hexmoor, H. (1994). *What are Routines Good For?* [On-line]. Available: hexmoor@cs.buffalo.edu.

Hexmoor, H. & Nute, D. (1992). *Methods for deciding what to do next and learning.* [On-line]. Available: hexmoor@cs.buffalo.edu.

Holldobler, B. & Wilson, E.O. (1990). *The ants*. Cambridge, MA: Belknap Press of Harvard University Press.

Howse, P.E. (1970). *Termites: A study in social behaviour*. London: Hutchinson & Co., Ltd.

Jaakkola, T., Jordan, M.I. & Singh, S.P. (1993). On the convergence of stochastic iterative dynamic programming algorithms. *MIT AI lab memo (No. 1441).*

Jaisson, P. (Ed.). (1983). *Social insects in the tropics*. Paris: Universite Paris-Nord.

Jantz, S.D, Yim, E., Harrison, R., Van Aken, R. & Doty, K.L. (1994, July). *Swarm robot materials handling in a manufacturing workcell*. *ALIFE4* Poster session presented at the annual meeting of Artificial Life IV, Cambridge, MA.

Jantz, S.D. & Doty, K.L. (1996). Learning to eat. *Proceedings of Robolearn 96: An International Workshop on Learning for Autonomous Robots, Key West, FL,* 46-53.

Jantz, S.D., Doty, K.L., Bagnell, J.A. & Zapata, I.R. (1997). Kinetics of robotics: The development of universal metrics in robotic swarms. *Proceedings of Florida Conference on Recent Advances in Robotics, Miami, FL.*

Kaufman, S. (1995). At home in the universe: The search for the laws of self-organization and complexity. Oxford: Oxford University Press.

Kelly, K. (1994). *Out of control: The new biology of machines, social systems, and the economic world.* New York: Addison-Wesley.

Kube, C.R. & Zang, H. (1994). Stagnation recovery behaviors for collective robotics. *Proceedings of Intelligent Robots and Systems Conference, Munich, Germany*, 1883-1890.

Kuniyoshi, Y., Riekki, J., Ishii, M., Rougeaux, S., Kita, N., Sakane, S., & Kakikura, M. (1994). Vision-based behaviors for multi-robot cooperation. *Proceedings of Intelligent Robots and Systems, Munich, Germany,* 925-932.

Levy, S. (1992). *Artificial life: The quest for a new creation*. New York: Pantheon Books.

Mahadevan, S. & Connell, J. (1992). Automatic programming of behavior-based robots using reinforcement learning. *Artificial Intelligence, 55*(2-3), 311-365.

Mahadevan, S. (1994). To discount or not to discount in reinforcement learning: A case study comparing R-learing and Q-learning. *Proceedings of the 11th International Conference on Machine Learning, New Brunswick, NJ,* 164-172.

Martin, F. & Wright, A. (Eds.). (1992). *The 6.270 robot builder's guide*. (Rev. ed.). Cambridge., MA: MIT Press.

Mataric, M.J. (1991). A comparative analysis of reinforcement learning methods. *MIT AI lab memo (No. 1322).*

Mataric, M.J. (1992). Minimizing complexity in controlling a mobile robot population. *Proceedings of IEEE International Conference on Robotics and Automation*, 830-835.

Mataric, M.J. (1994). *Interaction and intelligent behavior*. Unpublished doctoral dissertation, Massachusetts Institute of Technology.

Morley, D.W. (1954). *The evolution of an insect society*. London: George Allen and Unwin Ltd.

Parker, L. (1994). An experiment in mobile robotic cooperation. *Proceedings of the ASCE Specialty Conference on Robotics for Challenging Environments, Albuquerque, NM*.

Parker, L. (1995). *ALLIANCE: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots*. Oak Ridge, TN: Center for Engineering Systems Advanced Research, Oak Ridge National Laboratory.

Parker, L. (1993). Designing control laws for cooperative agent teams. *Proceedings of IEEE International Conference on Robotics and Automation, (Vol. 3)*, 582-587.

Reitman, E. (1994). *Genesis redux: Experiments creating artificial life*. New York: Windcrest/McGraw-Hill.

Resnick, M. (1994). *Turtles, termites, and traffic jams: Explorations in massively parallel microworlds*. Cambridge, MA: MIT Press.

Reynolds, C.W. (1994). Competition, coevolution and the game of tag. *Proceedings of Artificial Life IV, Cambridge, MA*, 59-69.

Seuss, Dr. (1965). *Green eggs and ham*. New York: Doubleday.

Skaife, S.H. (1961). *Dwellers in darkness*. Garden City, NY: Doubleday & Co.

Steels, L. (1993). The artificial life roots of artificial intelligence, *Artificial Life Journal, 1(1*).

Steels, L. (1994). Emergent functionality in robotic agents through on-line evolution. *Proceedings of Artificial Life IV, Cambridge, MA*, 8-14.

Terzopoulos, D., Tu, X. & Grzeszczuk, R. (1994). Artificial fishes with automous locomotion, perception, behavior and learning in a simulated physical world. *Proceedings of Artificial Life IV, Cambridge, MA*, 17-27.

Wang, J. & Premvuti, S. (1994). Resource sharing in distributed robotic systems based on a wireless medium access protocol [CSMA/CD-W]. *Proceedings of Intelligent Robots and Systems, Munich, Germany*,784-791.

Wang, Z., Nakano, E. & Matasukawa, T. (1994). Cooperating multiple behavior-based robots for object manipulation. *Proceedings of Intelligent Robots and Systems, Munich, Germany,* 1524-1531.

Walter, W. G. (1950). An imitation of life. *Scientific American*, 42-45.

Watkins, C. (1989). *Learning from delayed rewards*. Unpublished doctoral dissertation, Kings College.

Watson, J.A.L., Okot-Kotber, B.M. & Noirot, CH. (Eds.). (1985). *Caste differentiation in social insects*. Oxford: Pergamon Press.

Wilson, S.W. (1991). The animat path to AI. In. J.A. Meyer and S. Wilson (Eds.), *From Animals to Animats* (pp.15-21). Cambridge, MA: MIT Press.

Yagi, Y., Lin, Y. & Yachida, M. (1994). Detection of unknown moving objects by recipro-cation of observed information between mobile robots. *Proceedings of Intelligent Robots and Systems, Munich, Germany*, 996-1003.

Biographical sketch

I was born in Buffalo New York in 1971. From a young age I have been interested in robots and engineering. In 1982 my family relocated to Jacksonville Florida and I graduated as a National Merit Scholar in 1989 from Stanton College Preparatory School. I began my undergraduate degree at the University of Florida in the fall of 1989. In 1991 I began co-oping with Texas Instruments in Dallas, Texas. While at Texas Instruments I worked on soft error rates in semiconductor memories due to background radiation. I obtained three patents through my work at TI. During my semesters at UF I started working in the Machine Intelligence Lab under Professor Keith L. Doty. My work in robotics then continued into graduate school at the Machine Intelligence Lab.