Comparing the Performance of Heterogeneous and Homogeneous Swarms

Jason Alexander Hales

Follow this and additional works at: https://scholarsjunction.msstate.edu/td

Recommended Citation
Hales, Jason Alexander, "Comparing the Performance of Heterogeneous and Homogeneous Swarms" (2007). Theses and Dissertations. 1109.
https://scholarsjunction.msstate.edu/td/1109

This Graduate Thesis is brought to you for free and open access by the Theses and Dissertations at Scholars Junction. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Scholars Junction. For more information, please contact scholcomm@msstate.libanswers.com.
COMPARING THE PERFORMANCE OF HETEROGENEOUS AND HOMOGENEOUS SWARMS

By

Jason Alexander Hales

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Masters
of Computer Science
in the Department of Computer Science

Mississippi State, Mississippi

December 2007
COMPARING THE PERFORMANCE OF HETEROGENEOUS AND HOMOGENEOUS SWARMS

By

Jason Alexander Hales

Approved:

Susan M. Bridges
Professor of Computer Science
(Major Professor)

Julia E. Hodges
Professor of Computer Science
(Committee Member)

Andrew B. Watkins
Instructor of Computer Science
(Thesis Director)

Edward B. Allen
Associate Professor
(Graduate Coordinator)

W. Glenn Steele
Dean of the Bagley College of Engineering
This thesis compares the performance of heterogeneous and homogenous swarms. Swarms are defined as particles or agents which react to their environment and fellow particles or agents according to social rules. Three attributes weights of an individual agent were varied for these experiments: Collision Avoidance with individual agents in the swarm, Center of Mass of the swarm and the parameter that controls Velocity Matching in the swarm. In homogenous swarms, all individuals had the same attribute weights while in heterogeneous swarms weights for one attribute were taken from a normal distribution for the population. Results show that heterogeneous swarms outperformed homogenous swarms if the weights for the Center of Mass Weight attribute were heterogeneous in the population. The Collision Avoidance and Matched Velocity attributes showed little performance difference for heterogeneous and homogenous swarms. However, swarms heterogeneous in the Matched Velocity parameter showed substantial performance improvements for the most difficult map.
DEDICATION

I would like to dedicate this research to my family (Kristina, Drew and Aidan).
ACKNOWLEDGMENTS

The author wishes to express thanks to Eric for watching my swarms with an inquisitive mind. I want to thank the Santa Fe Institute for providing the swarm framework used in my research. Lastly, I want to thank Dr. Bridges and Dr. Watkins for sticking with me through the long process of doing this thesis.
# TABLE OF CONTENTS

**DEDICATION** .......................................................................................................................... ii

**ACKNOWLEDGMENTS** ............................................................................................................ iii

**LIST OF TABLES** ...................................................................................................................... vi

**LIST OF FIGURES** .................................................................................................................... vii

I. **INTRODUCTION** ................................................................................................................... 1
   1.1 Objective of Thesis ............................................................................................................. 2
   1.1 Thesis Outline .................................................................................................................... 3

II. **LITERATURE REVIEW** ....................................................................................................... 5
   2.1 Defining Swarm Intelligence ............................................................................................. 5
   2.2 Application of Swarm Intelligence .................................................................................. 9
   2.3 Related Fields ................................................................................................................... 11
   2.4 Homogenous and Heterogeneous Swarms ...................................................................... 13
   2.5 Conclusion ....................................................................................................................... 16

III. **SIMULATION ENVIRONMENT AND SWARM ALGORITHM** ........................................... 18
   3.1 Simulation Environment ................................................................................................... 18
   3.2 Terrain and Tasks .............................................................................................................. 20

IV. **EXPERIMENTAL DESIGN** ................................................................................................ 30
   4.1 Preliminary Experiments .................................................................................................. 31
   4.1.1 Validation Experiment Results .................................................................................... 31
   4.1.2 Swarm Initialization ..................................................................................................... 33
   4.1.3 Approach for Establishing Reasonable Weight Ranges ............................................. 34
   4.1.4 Preliminary Heterogeneous Versus Homogeneous Experiment .................................. 36
   4.1.5 Heterogeneous Versus Homogenous Swarms for Near-Stalemate Weights ............... 38
   4.2 Examples of Swarm Movement ....................................................................................... 40
   4.3 Experimental Design for Further Experiments ............................................................... 45
LIST OF TABLES

3.1 Description of variables used in update algorithm ..................................................28

4.1 Attribute Weights for Preliminary Experiments ...................................................32

4.2 Experiment Parameters ........................................................................................46

5.1 Center mass weight performance for both heterogenous and homogenous swarms ..........................................................................................................................51

5.2 Student T-Test Results Comparing Homogeneous and Heterogeneous Swarms for Center of Mass Attribute Weight ..............................................................53

5.3 Collision Avoidance Weight performance for both heterogenous and homogenous swarms. ..............................................................................................................55

5.4 Matched Velocity Weight performance for both heterogenous and homogenous swarms. ..............................................................................................................57
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Class diagram of custom classes with attributes and methods exposed.</td>
</tr>
<tr>
<td>3.2</td>
<td>The six maps used in these experiments.</td>
</tr>
<tr>
<td>3.3</td>
<td>Example particle trace on map 6.</td>
</tr>
<tr>
<td>3.4</td>
<td>Update algorithm for agent’s velocity vector and location at each time step (adapted from [21]).</td>
</tr>
<tr>
<td>4.1</td>
<td>Validation experiments where the weight values used where the legend gives the values used for $&lt;w_c, w_{mv}, w_a&gt;$.</td>
</tr>
<tr>
<td>4.2</td>
<td>Winder and Reggia’s [21] results where the legend gives the values used for $&lt;w_c, w_{mv}, w_a&gt;$.</td>
</tr>
<tr>
<td>4.3</td>
<td>Map 4 showing stalemate of swarm at the first hazard.</td>
</tr>
<tr>
<td>4.4</td>
<td>Performance of the swarms with different Center of Mass weight values on Map 2 and Map 4. Homogenous swarms with $w_c$ values from 0.04 to 0.40 are compared to heterogeneous swarms.</td>
</tr>
<tr>
<td>4.5</td>
<td>Comparison of the speed and the number of collision occurring in the swarm as the Center of Mass weight is varied (all swarms are homogeneous).</td>
</tr>
<tr>
<td>4.6</td>
<td>Mean performance between homogenous and heterogeneous swarms when center mass is between 0.39 and 0.42 on Map 2.</td>
</tr>
<tr>
<td>4.7</td>
<td>Agent routing around the mountain terrain in Map 2 by exploring and following fellow agents around the hazard. Mountains are purple, swamp is gray, goal is red, agent path is green and white is open terrain.</td>
</tr>
<tr>
<td>4.8</td>
<td>Agent routing around multiple terrain hazards in Map 5. Mountains are purple, swamp is gray, goal is red, agent path is green and white is open terrain.</td>
</tr>
</tbody>
</table>
4.9 Map 5 hazards demonstrating an approximation of the u-shaped hazard researched by Wan and Chen [32].

4.10 Map 1 hazard of a large swamp, the agents veer around the swamp only touching the edges as it move to the next goal.

4.11 Map 3 hazards where the agents wandered around the top edge to find the right-side. Note how close the agents could of split with some going right and others going left.

4.12 Map 3 showing the completed path of the swarm about the hazards.

4.13 Map 1 demonstrates agents on the move; they are the blue dots in the swamp.

5.1 Performance of swarms for varying values of the Center of Mass weight.

5.2 Collision Avoidance Weight performance for both heterogenous and homogenous swarms.

5.3 Matched Velocity weight performance for both heterogenous and homogenous swarms.
CHAPTER I
INTRODUCTION

Swarm systems are collections of simple computational agents (agents and particles are used synonymously throughout this thesis) that exhibit self-organizing collective movements based on individual interactions with the environment and local interactions among the particles. Although these systems have no centralized control and each autonomous agent is typically a simple reflexive agent, the collection of agents often exhibits interesting emergent behavior that can be viewed as intelligent. A number of biological systems have served as models for these systems including flocks of birds [1], social insect swarms [2], schools of fish [3], and colonies of bacteria [4]. Computational models of these systems have been widely studied in recent years in order to gain a better understanding of the biological systems upon which they are based and to take advantage of the capabilities of the resulting computational systems. For example, particle systems (and related methods) have been studied for use in a number of applications including mine detection [5] [6], computer graphics [7], clustering [8], numerical optimization [9], sensor networks [10], and space exploration [11].

The individuals in particle systems each have a number of attributes (or parameters) that determine how the particle moves in the environment, how it responds to obstacles in the environment, and how it interacts with other particles. Particle swarms
are often modeled as groups of homogeneous agents that differ only in location and direction of movement. Although there has been substantial work in generation of sets of diverse agents for goal finding and for optimization through machine learning and evolutionary methods [12] [13] [14], there has been little systematic study of the effects of heterogeneity on the performance of particle systems.

An agent’s behavior is determined by its reaction to information (stimulus). This behavior is determined by the internal equations and algorithms of a given agent. The agent’s behavior determines how the agent will navigate its territory and complete its goals. If an agent does not respond appropriately to stimuli, it may be unable to complete its tasks. For example, Goldenstein encountered such a phenomenon while varying the attributes of the agents [15].

This goal search problem of agent behavior may also be viewed as an optimization problem where the goal is to find the global minima. If an agent with one sort of behavior becomes trapped in a local minimum, say a terrain hazard, then we hypothesize that another agent with different behaviors might be able to overcome this local minimum. This type of behavior was identified by Buffet et al. in their discussion on agents becoming trapped in local minima [16]. This raises the question: can diversity in a population of agents enable a population to overcome becoming trapped in local minima and discovering a global minima.

1.1 Objective of Thesis

The goal of this thesis research is to determine the effects of heterogeneity on the performance characteristics of swarms in terms of the number of steps required to reach
the goal and the difficulty of the problems it can solve. We have investigated the influence of heterogeneity of the weights that control the importance of the visual range of the agent, that enable a particle to match the velocity of its neighbors, Collision Avoidance behavior, the influence of the Center of Mass of the neighbors, and the influence of the goal destination. In preliminary experiments we have shown that heterogeneity has little influence on the performance of the swarm if the mean value for the attribute weights is near the optimum for a particular problem. However, when the mean attribute weights are near the extremes of the acceptable ranges, the heterogeneous swarms can solve problems in fewer steps than the homogeneous swarms and can sometimes complete problems that the homogeneous swarms cannot solve. Based on these results we developed the following hypothesis:

Compared to swarms with homogeneous weights for attributes, swarms with heterogeneous weights for attributes can, when the mean weight is near the margin of acceptable values, 1) reach the goal in fewer steps and 2) solve some problems that homogeneous swarms cannot solve.

1.1 Thesis Outline

The remainder of this thesis is organized as follows:

- Chapter II provides a review of literature relevant to the domain of particle swarms and particle swarm optimization;
- Chapter III describes the mathematical model and simulation environment used in this study;
- Chapter IV describes the experimental methods used to test the hypothesis;
• Chapter V describes the results of experiments comparing the performance of homogeneous and heterogeneous swarms;

• Chapter VI provides the conclusions to this thesis and future work related to the thesis.
CHAPTER II
LITERATURE REVIEW

This chapter presents an overview of previous work on swarms and swarm intelligence. It begins by presenting a definition of swarm intelligence and examining some of the fundamental concepts important to the development of swarm behavior for computational purposes. Next, this chapter examines various uses of swarm intelligence for problem solving. Following this is a discussion of related fields of study. The chapter concludes with an argument concerning the need for diverse behavior from individuals in the swarm.

2.1 Defining Swarm Intelligence

The inspiration for the use of swarms for computational problem solving came from observations of naturally occurring phenomena. For decades, researchers have been fascinated with the group behavior exhibited by such animals as social insects (for example, ants, bees and termites) and higher-order animals such as birds, fish, and sheep. These groups of animals are made up of individuals. However, these individuals interact in such a way to exhibit global behavior that is distinct from the actions of any one individual in the hive, nest, herd, or school [1]. In general, swarms can be characterized as particles or agents that react to a given stimuli in their environment including reacting
to other agents. Swarm intelligence is the emergence of global, intelligent behavior from
the interactions of self-organizing individuals within the swarm [17]. This section
discusses some of the natural inspiration for swarm intelligence for computational
problems. It presents the importance of the collaboration of individuals for global
behavior. This collaboration introduces the concept of the emergence of intelligent group
behavior.

Bonabeau [2] uses the analogy of social insects as the basis for Swarm
Intelligence. The starting point for discussing Swarm Intelligence is a combination of
structural and behavioral items of the agents and the overall environment. Social insects,
such as ants, wasps, and some termites, know little or nothing about the global
environment in which they live; they do have knowledge about the environment from
their local perspective. Thus ants have no global road-map of what is an optimal path
would be. Ants use stigmergy to determine the path to follow by referencing previous
ants’ pheromones. The analogy to Swarm Intelligence occurs because even if there is
only one ant on a given path it has a reference to the previous ants’ paths and so behaves
as though it is in a crowd of ants. Swarm intelligence occurs when an ant takes a slightly
different path than a previous ant and thus makes a different path, a possibly more
optimal path, than the previous one.

One of the key traits about these natural phenomena is the ability to find optimal
solutions in a decentralized manner. For example, collectively ants are able to find
optimal paths to food with only local reactions of individuals to the environment through
pheromones [2]. This self-organizing property of stigmergy leads to the idea of group
intelligence emerging from the individual behavior [2] [18] [19]. This decentralized control structure is in contrast to much of what has been previously studied in computer science. Observation has led to, for example, models of robot control that individually embody simple rules but collectively exhibit optimal path-finding behavior [20]. Social insects also exhibit natural clustering or sorting behavior by particular nests of individuals grouping together and away from other nests [2] [8]. This is also useful for computational studies where clustering of data is of importance [11].

Computer scientists have been most interested in understanding this global behavior. In an attempt to more fully model this form of global behavior, computer scientists have defined simple rules that individuals in a swarm must follow [21]. With each individual following these rules, the global intelligence of the group is allowed to emerge [2] [17]. Beni and Wang, in their seminal work involving small robot swarms referred to as Distributed Robotic Systems (DRS), sum up Swarm Intelligence best with the “problem is to design a system that, while composed of unintelligent units, is capable, as a group, to perform tasks requiring intelligence” [17].

One of the most cited references on swarms is Reynold’s paper “Flocks, Herds, and Schools: A Distributed Behavioral Model” because of its innovation in using computer graphics to simulate the natural phenomenon of flocking behaviors [1]. Although the paper is not on Swarm Intelligence, it allowed the outside observer to witness the concepts of cooperation and Collision Avoidance. Distributed behavior is where the “boid” [1] is given a set of behaviors and is allowed to interact with the environment based on its own local perspective. Reynolds’s gives a comparison of
“boid’s” to particle systems by explaining that boids have a shape and direction “orientation” [1]. Complexity as cited by [2] is expressed as a differentiator of [1] boids to particle systems of the day. Reynolds’s paper is also important in that it defines some of the key aspects of the algorithm for implementing simulated swarms. These are “collision avoidance,” “velocity matching,” and “flock centering” along with the concept of “maximum acceleration.” These three aspects of the algorithm are used in adjusting the behavior of an agent by updating the vector used by the agent to guide it to its goal. “Collision avoidance” along with goal seeking yields the competitiveness of the agents in the swarm. “Velocity matching” and “flock centering” implement the cooperative behavior of the agent to the swarm.

The ability to find optimal solutions in a decentralized manner is directly related to the idea of self-organization within these natural phenomena. One of the more fascinating realizations about social insects and these other animals is that though there is no global control, individual reactions to the environment lead to organized behavior of the group. Within the group, individuals can react in a cooperative or a competitive manner [19]. Both of these types of behaviors can lead to different group organization. However, the key point here is that through an individual’s reaction to the environment, the individual changes the environment, which in turn changes the reactions of other individuals to the environment. A series of simple individual changes and reactions leads to the emergence of global behavior.

Swarm Intelligence draws its strength from the interactions among the agents. The behavior of the agents creates the dynamic of different swarms. Elliott and Kiel [18]
proposes combining all agent behavior into only cooperation and competition between the agents. Although it is possible to implement a swarm based on only two behaviors, this configuration does not allow detail investigation into the dynamics of Swarm Intelligence performance. Mataric [22] gives the insight into the “minimal set of agent needs to reach its goal repertoire.” The proposed set of behaviors is defined as “avoidance, following, aggregation dispersion, homing and wandering”. This constitutes the basic set of behaviors associated with a group. Mataric [22] also uses “heterogenous reward functions” by creating sub-goals for each goal. This is used to create multiple goal points for a swarm to route itself though a given map. Mataric [22] does not address the learning by individual agents and thus does not reap the full benefit of intermediate learning reinforcement by the agent at each sub-goal.

This section has focused on what swarm intelligence means. In general, swarm intelligence is concerned with the global behavior that emerges from the interactions of fairly simple-minded individuals. One of the key properties of swarm intelligence from a computational perspective is the decentralized nature of its control structure. This allows computer scientists to program fairly simple rules into individual components whose interactions lead to interesting global behavior. In the next section, examples of how swarm intelligence has been used for problem solving are presented.

2.2 Application of Swarm Intelligence

With some basic understanding of what Swarm Intelligence is, we now turn our attention to how these concepts have been applied to solving real world problems. This
section presents a selection of exemplars in the application of Swarm Intelligence to a variety of domains.

Swarm Intelligence has been applied in improvements to the detection of landmines in a combat zone. Kumar and Sahin [6] applied the principles of social insects’ (ants), collective transportation capabilities to quickly identify and defuse landmines at random locations. Although, there are not robots currently with this capability, the research was able to simulate an approximation of the results if actual landmine robots existed [6] and also characterized a problem where a swarm is in a frozen state. A frozen state occurs when an ant swarm stays at an intermediate location indefinitely and is unable to move to the next location. This thesis identifies frozen state as stalemate.

Clustering of data using an ensemble of swarms with different capabilities has been applied using swarm intelligence. Yang and Kamel [8] used three ant swarms with unique speed models of constant, random and randomly decreasing. Different ant swarms were then combined in an ensemble to develop the clustering of differing data collections. Generally the results of [8] gave improvements when using the ensemble over the individual swarms. These results provide an indication of the relative performance of homogenous and heterogeneous swarm performance for an applied problem where heterogeneous is defined as an ant swarm ensemble where speed is the only attribute being varied.

Complex simulation of crowds has been developed by using agent-based simulation with techniques similar to swarm intelligence. An excellent example is [23]
where Collision Avoidance was used to create autonomous crowds for visual graphic applications. The crowds (i.e. Swarm) has two types of goals, static and dynamic, where the dynamic goal is adjusted by the avatar’s location. The avatar is a member of the crowd given specific instructions by the researcher.

Use of Swarm Intelligence has also been applied to exploring production system optimization. Grobler, Engelbrecht, Joubert, and Kolk [24] used a swarm intelligence technique to determine production scheduling with material and equipment availability. The production system had to account for many system constraints. These constraints included release dates, no intersection of operations and precedence relationship of produced items. Although they [24] did not achieve the optimization hoped for, the research demonstrated how swarm intelligence can be used to address complex issues.

Swarm Intelligence has been used to layout faculty locations based on categorizations. Hardin and Usher [25] applied swarm intelligence to individual faculty as square tiles to self-organize according to department affinity. Self-organization is one of the emergent behaviors that occur from swarm intelligence with no central control. Development constraints in representing their data as square tiles instead of individual offices actual size limited their results. The results by Hardin and Usher [25] using the swarm technique where “consistent with other available algorithms”.

2.3 Related Fields

Swarm Intelligence is part of a larger multi-agent field with many different sub-fields. Multi-agent systems with behaviors could be considered a subset of Sociology. One closely related field is Particle Swarm Optimization (PSO) which uses Swarm

11
Intelligence for function optimization. PSO has been successful in reaching global minima with performance better than Evolutionary Algorithms. Ant Colony Optimization (ACO) [26] uses stigmergy as the primary mechanism for optimizing for the shortest path optimization. Others have modified the PSO to optimize the performance with a given set of benchmarks.

A slightly different implementation of Swarm Intelligence is Particle Swarm Optimization. PSO uses a global or neighborhood best to inform all in the swarm to direct themselves to this more global best fit. PSO uses the simulation step approach to incrementally progress through its iterations; this is needed to update to best values to each of the particles. PSO then takes this new vector for each particle and introduces a linear randomization so that the swarm does not stalemate on specific local minima. PSO has shown promise in determining optimization for a given fitness factor. PSO has well-defined heuristics for setting particle weights in a swarm. Although PSO is often compared to evolutionary algorithms, its underlying approach is one of Swarm Intelligence based on observations of social insects. PSO states this as “the real strength of the particle swarm derives for the interactions among particles as they search the space collaboratively” [27].

Gaussian Particle Swarm Optimization (PSO) [14] leverages the PSO algorithm by creating a Gaussian instead of a linear distribution of the inertia weight, resulting in increased performance. The Gaussian PSO increases its performance by having more particles in the mid range of the distribution. Thus more of the swarm follows the previous best particle’s values as opposed to the linear distribution from zero to
maximum. Although the Gaussian PSO results indicate a higher performance than standard PSO on “well-known benchmark functions” it is does not validate if the Gaussian or linear is a more generalized method for all functions. Failure to reach global optimization, local minima, was not part of experiment for either the Gaussian or Linear distribution.

ACO [26] is a technique where the pheromones left by ant are reduced in potency over time. If the ant searches a given space and does not return or is slow in returning then the pheromones trail is reduced for the next ant. If on the other hand the return of an ant is short and its pheromone trail is refreshed then its potency is still relatively high. This, in turn, will increase the number of ants on that trail and a shorter trail is much more active with pheromones than a path which is longer and spread out. This relates to Swarm Intelligence by having a swarm searching for a more optimal path [2].

2.4 Homogenous and Heterogenous Swarms

For this thesis, heterogeneity and homogeneity are restricted to weights (importance) of the individual agent/particle behavior attributes [22]. The majority of the studies in the field of Swarm Intelligence have been concerned with homogenous swarms. In a field related to Swarm Intelligence, heterogeneous groups of robots have been viewed having advantages over homogenous groups [28]. Homogenous crowds are less efficient than heterogeneous ones [29]. Heterogeneous agents have shown indications of improved performance in overcoming local minima issues [16]. This thesis explores the possible performance differences between heterogeneous and homogenous particle swarms.
A key reference for heterogeneous performance is the paper by Hamagami and Hirata [29] on “Method of crowd simulation using multiagent on cellular automata” where they compared homogenous and heterogeneous multi-agent simulations. Hamagami and Hirata [29] and other crowd simulation research [23] are similar to swarm research in that it is multi-agent, set behaviors, set physical properties, and in a goal oriented environment. The volume associated with each specific agent created collisions with other agents as the crowd moved through the environment. The cellular automata plane used by Hamagami and Hirata [29] is equal to this thesis’s Swarm’s tool kit of a grid system for map movement. As the agents move about the environment they have to plan/react to other agents and their environment from their own perspective. The key results from the paper indicate that modeling a crowd with homogenous agents make “whirlpool, waves, and blanks” [29] and generally are slower to reach their goal, where heterogeneous multi-agent crowds “formed lines, then flows efficiently” [29]. This thesis attempts to design experiments to investigate this phenomenon and determine actual performance differences between homogenous and heterogeneous multi-agent crowds; the supposition is that heterogeneous multi-agent crowds perform goal search faster.

Rouff, Vanderbilt, Hinchey, Truszkowski, and Rash [11] state that one of the main advantages of Swarm Intelligence is “homogeneous swarms, due to their differing environments, may learn different things”. Is the emergence of intelligence in a swarm related to the advantage of having multiple perspectives to search out more optimal paths? Rouff, Vanderbilt, Hinchey, Truszkowski, and Rash [11] state that the difficultly
of determining the emergence of swarm intelligence is due to the complexity of a “huge state space”.

In Beni and Wang’s paper [30] the authors identify the “most urgent problems” with DRS. One of these problem identified was with multiple physical robots engaged in a single goal. This problem has been characterized as stalemate or “trap-avoiding” [30]. This problem manifests itself by having agents continuously colliding in an attempt to reach the same point. From an optimization point of view this is a local minimum. Local minima from Beni and Wang’s paper [30] perspective could be overcome by having a traffic control schema for the agents. Beni and Wang’s [30] example of “mentally handicapped painters” attempting to leave the room at the same time is encountered when all of the painters have the identical characteristics. If, for example, the painters had a diversity of characteristics, would that change the problem of being confined in a local minima and would the behavior [29] identified where the painters would form lines and exit successfully be exhibited?

Another approach to overcome local minima is to modify the behaviors of the agents as needed. Buffet, Dutech, and Charpillet [16] used a reward method for behaviors by recombining of basic behaviors and then modifying the agent’s behaviors. There is a computational cost as iterations of behavioral change require some processing of existing behaviors and proposed newer ones.

Evans, Unsal, and Bay [28] state “homogenous swarms, which are composed of similar robots, have many advantages over heterogeneous systems.” Examples of the advantages are not from a performance perspective of the swarm but are related to
interchangeability of the agents/robots, decentralized nature of the robots, and ease of manufacturing homogenous robots. All of these could be true or they could be used as rationale for homogenous agents/robots. Bonabeau, Dorigo, and Theraulaz [2] stated that as one of the “even more fundamental than the issue of programming the system, is that of defining it … Should they be all identical?”

Science does not have a direct answer for why there is diversity in nature. It does show some interesting consistencies for physical characteristics to be normally distributed. Some fundamental questions arise as to why nature has diversity except as an artifact of evolution. Suppose nature uses diversity in physical characteristics in the same way as Swarm Intelligence uses agents to optimize. Suppose an agent, which is in failure—a local minimum, is able to collaborate with an agent which is not in failure. For example, suppose a tall and short person were traveling together. As they travel down a path the shorter person could identify hazards and path performance from his perspective. On the other hand the taller person would do the same. Without communicating verbally they could collaborate by watching each other advance through a series of obstacles in a way that a group of people of identical heights cannot.

2.5 Conclusion

This thesis chapter has reviewed some of the literature relevant to the study of Swarm Intelligence. It examined a variety of definitions for this phenomenon, all of which focus on the emergence of global intelligent behavior through the cooperation and competition of simple local agents. A few applications of this phenomenon in the computing field were discussed as well as fields related to Swarm Intelligence. Finally,
in keeping with the purpose of this thesis, the potential benefits of heterogeneous populations of agents were posited. The remainder of this work focuses on the question of the benefit of heterogeneity in particle swarms.
CHAPTER III

SIMULATION ENVIRONMENT AND SWARM ALGORITHM

The apparatus used in these experiments is a virtual environment where particles move through a simulated terrain in search of goals and in the process react with each other and with their environment. These experiments are similar to those described by the Winder and Reggia study [21] on the effects of distributed memory in particle movement. Unlike Winder and Reggia, we have not incorporated particle memory. In addition, the agents in our simulation environment occupy space while those used by Winder and Reggia do not.

3.1 Simulation Environment

The simulation environment for these experiments was constructed from the Swarm toolkit available at www.swarm.org. The toolkit was originally developed by the Sante Fe Institute [31] and is now distributed by the Swarm Development Group. The foundation of this toolkit provides a set of object-oriented libraries enabling implementation of a wide variety of agent-based simulations. The libraries provide tools for layering environments, particle definitions, and simulation control. Our implementation for this particular swarm simulation environment used the toolkit’s Java interface [31]. We also used NetBeans for development, verification, and validation.
This simulated environment is step-based so discrete events happen on a uniformly segmented timescale. The environment is implemented with layers for maps, agents, and goals. The layering allows objects of different layers to reside in the same location. Simulation controls include starting, stopping, and data recording. The simulation environment has been implemented for running in batch to facilitate easier experimentation. Configuration files were used to allow experiment parameters to be defined.

The implemented architecture includes the Swarm toolkit referenced earlier plus the author’s classes expressed in the Unified Modeling Language class diagram of Figure 3.1. Figure 3.1 shows the methods used by each particle (called a Traveler), an agent with specific goals, to move about the terrain. The following is a description of the author’s classes and their basic functionality:

- Vector is an abstract class for the definition of vectors as attributes.
- StartTraveler initiates and ends the swarm engine and passes the experiment to the swarm engine.
- SwarmUtils is used by Swarm to overcome issues related to using Java and Objective C related to multi-language software. This class was not changed from the original sample given by the swarm toolkit.
- MapSpace manages the loading of the terrain maps and goals for the swarm engine.
- ModelSwarm defines initial values used by the experiment and travelers for a given swarm run. ModelSwarm also manages the discrete steps used by swarm to run the experiment.
• ObserverSwarm is used to manage the swarm engine’s performance as it runs the experiments. Additionally, it manages the user interface to the swarm engine of controlling the experiment.
• Traveler is the agent used by these experiments to move through the terrain. The traveler class includes methods for calculating the agent’s velocity based on the given environment and situation.

Figure 3.1 Class diagram of custom classes with attributes and methods exposed.

3.2 Terrain and Tasks

The Swarm simulation environment implements a virtual square plane with a two-dimensional grid superimposed with terrain obstacles (swamps and mountains). This plane is henceforth referred to as the map. The maps used for our experiments are 1,280
Individual particles are the same size as a single map square and pursue a set of goals by moving between goal coordinates. Each grid square is only allowed to contain one particle at a time and thus collisions are possible. Individual goals are located at different places on the map. The overall system performance goal is defined as moving “collectively from goal to goal in the shortest time possible” [21]. When 90% of a swarm has reached the current goal, the swarm is given the next goal.

For this thesis, we have used six different terrain maps of varying terrain and goal layouts. The basic layout of these maps is modeled after the maps described in Winder and Reggia [21] for their experiments. Figure 3.2 shows the different maps. The maps use color to designate items as follows:

- **White** – Open terrain, the easiest terrain to traverse.
- **Light Gray** – Swamp terrain, similar to open terrain but slightly slower.
- **Purple** – Mountain terrain, hardest terrain for particles to traverse.
- **Red** – Goal location
- **Blue** – Particle
- **Light Green** – Particle path trace
Figure 3.2 The six maps used in these experiments.

Map 2, Map 3, and Map 4 can be categorized as having a square mountain with some swamp on the sides between each of the goal waypoints. Map 1 has a square swamp between each goal waypoint. Map 5 has large square swamps as well as fields of small square mountains. Map 6 has square mountains and swamp hazards as well a field of random square mountains. Figure 3.3 shows an example of the particle trails left behind as a simulation is running and illustrates that the particle swarm tends to avoid slower terrain where possible.
The algorithm for particle movement and interaction is made up of 3 basic steps. The first step is to determine the influences of the goal location and fellow particles on a given particle’s velocity. The second step is to update the new velocity based upon the terrain and maximum velocity allowed. In the third and last step, the particle’s position is updated unless a collision with another particle has occurred.

The state of an agent at a particular time step can be described by its location ($x_j$) and its velocity vector ($v_j$). The velocity vector determines where the agent moves at the next time step. During each step, the velocity is updated using the algorithm shown in Figure 3.4. At the beginning of the simulation, each agent is given information about the location of all goals and the sequence in which they are to be visited. At each time step,
the velocity vector is recomputed based on the current value and the weighted influence of the goal destination, a Collision Avoidance parameter, the Center of Mass of the neighbors, a Matched Velocity parameter, and the number of obstacles in view. After the velocity vector is updated, the influence of local obstacles is taken into account. Movement is slowed in mountains and in swamps. The values of the parameters $k_s$ and $k_r$ determine the extent to which mountains and swamps impede movement. Map boundaries also influence movement.

The following is a breakdown of the individual equations in Figure 3.4.

Equation 3.1 computes the goal vector. The goal vector gives the direction to the goal along with an influence weight. Goal locations are preset and overlaid on the map before the experiment is run. When 90% of the particles reach their given goal, all particles begin to pursue their next goal or the simulation is complete. This gives one particle out of ten the opportunity to pursue their next goal without reaching their current one.

\[
v_g = \frac{x_g - x_j}{|x_g - x_j|}
\]  

(3.1)
For agent $j$ with current position $x_j$ and velocity $v_j$

- $M$: number of agents within radius $r_{tc}$ of agent $j$; Other agents too close to $j$
- $N$: number of agents within radius $r_{vr}$ of agent $j$; Other agents in view of $j$
- $O$: set of all grid locations within radius $r_{vr}$ of agent $j$; Viewable terrain in view of $j$
- $P$: number of $O$ elements for a given agent $j$; Number of grid squares within view of $j$
- $g$: coordinates of current goal cell for all agents; Goal location of agent $j$
- $u$: unit vector from agent $j$ towards a specific $O$; Signed values of the view
- $k$: terrain value of a specific grid location; Number of grid squares within view of $j$
- $s$: mountain or swamp terrain value; Mountain and swamp $v_{max}$
- $max$: Maximum speed
- $tc$: too close to neighbors radius; Comfort zone
- $vr$: visual range of an agent; How far can an agent see

**Step 1:** Compute influences on agent

\[ v_g = \frac{(x_g - x_j)}{|x_g - x_j|} \] ; Influence of goal destination

\[ v_a = \frac{\sum_{i=1}^{M} (x_j - x_i)}{M} \] ; Collision Avoidance

\[ v_c = \frac{\sum_{i=1}^{N} (x_j - x_i)}{N} \] ; Center of Mass of neighbors

\[ v_{mv} = \frac{\sum_{i=1}^{N} v_i}{N} \] ; Matched Velocity of neighbors

\[ v_{vr} = -\sum_{x \in O} (u_x k_x) / P \] ; Direction with the slowest obstacles in view

\[ v_j = v_j + w_g v_g + w_c v_c + w_{mv} v_{mv} + w_a v_a + w_{vr} v_{vr} \] ; New aggregate velocity for agent

**Step 2:** Adjust velocity

\[ \text{if } v_j > v_{max} \text{ then } v_j = v_{max} / |v_j| \] ; Velocity stays below $v_{max}$

\[ v_j = v_j \left( \frac{2v_{max} k_x |v_j|}{|1 + e^{v_j - v_{max}}|} \right) \] ; Slow down if in a swamp/mountain

**Step 3:** Update position

\[ x_j = x_j + v_j \]

Figure 3.4 Update algorithm for agent’s velocity vector and location at each time step (adapted from [21]).
Equation 3.2 computes the Collision Avoidance vector; an averaged group vector, excluding the current particle, is created in the opposite direction along with an influence weight. This causes particles to be influenced in the opposite direction of the group to avoid collision. Particles can still have collision because each grid square can contain only one particle.

\[
v_a = \frac{\sum_{i=1}^{M} (x_j - x_i)}{M}
\]

(3.2)

Equation 3.3 computes the average direction of the particle group, excluding the current particle. This causes the particle to match the general velocity of the swarm.

\[
v_{mv} = \frac{\sum_{i=1}^{N_{agent}} v_i}{N_{agent}}
\]

(3.3)

Equation 3.4 computes a vector to the center of the swarm, excluding the current particle, along with an influence weight. This vector causes the particle to move toward the center of the mass of particles. When the results of equations 3.2, 3.3, and 3.4 are combined they give the group dynamic of flocks and schools as seen in Reynolds [1].

\[
v_c = \frac{\sum_{i=1}^{N_{agent}} x_i - x_j}{N_{agent}}
\]

(3.4)

Equation 3.5 represents the particle’s visual ability to gain knowledge about its environment. During the simulation particles use their vision to identify terrain, fellow
particles, and goal locations. Equation 3.5 is a simple technique that allows each particle to look around itself to view the contents of all grid locations within a certain radius defined in units of grid squares away from the particle.

\[ v_{vr} = -\sum_{x \in O} (u_{x}, k_x) / P \]  

Equation 3.6 computes the weighted sum the vectors to form the influence vector of the particle. The influence weights define the individual particles reactive behavior. The influence vector is added to the particle’s current vector.

\[ v_j = v_j + w_g v_g + w_c v_c + w_{mv} v_{mv} + w_a v_a + w_{vr} v_{vr} \]  

Inequality of equation 3.7 and equation 3.8 are used to govern the maximum speed of a particle. Inequality 3.7 determines if equation 3.8 is applied to the particle’s velocity vector where equation 3.7 is applied to slow the particle to within the maximum speed.

\[ |v_j| > v_{\text{max}} \]  

\[ v_j = \frac{v_j v_{\text{max}}}{|v_j|} \]

Equation 3.9 applies terrain effects to the particle’s velocity vector. If the particle is in a mountain or swarm grid square then Equation 4.9 is applied to reduce the speed of the particle. These effects are defined by Winder and Reggia:

“In swamp cells, the maximum velocity drops by a factor of 0.5, cutting an agent’s speed in half. In mountain cells, the maximum velocity drops by a factor of roughly 0.03, brining an agent to a speed so slow that it is ineffective.” [21].
Table 3.1 Description of variables used in update algorithm

<table>
<thead>
<tr>
<th>Equations Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_g )</td>
<td>Coordinates of the current goal grid location.</td>
</tr>
<tr>
<td>( x_j )</td>
<td>Coordinates of the current agent.</td>
</tr>
<tr>
<td>( M )</td>
<td>The collection of agents inside the comfort range.</td>
</tr>
<tr>
<td>( N_{agent} )</td>
<td>The collection of agents inside the visual range.</td>
</tr>
<tr>
<td>( N_{terrain} )</td>
<td>The collection of terrain inside the visual range.</td>
</tr>
<tr>
<td>( k_t )</td>
<td>The constant grid square terrain value: 0 = open, 0.05 = mountain, 0.2 = swamp, 0.203 map boundary, -0.17 current goal.</td>
</tr>
<tr>
<td>( k_s )</td>
<td>The modifier of max speed in different terrains: 0.5 = swamp, 0.03 = mountain.</td>
</tr>
<tr>
<td>( w_g )</td>
<td>Weight value to influence the goal vector of the current goal.</td>
</tr>
<tr>
<td>( w_c )</td>
<td>Weight value to influence the center mass vector caused by the collection of agents inside the visual range.</td>
</tr>
<tr>
<td>( w_{mv} )</td>
<td>Weight value to influence the matching of the collection of agent’s velocity inside the visual range.</td>
</tr>
<tr>
<td>( w_a )</td>
<td>Weight value to influence the Collision Avoidance vector caused by the collection of agents inside the comfort range.</td>
</tr>
<tr>
<td>( w_{vr} )</td>
<td>Weight value to influence the visual terrain vector caused by the collection of terrain inside the visual range of the agent.</td>
</tr>
<tr>
<td>( v_g )</td>
<td>The goal vector too the current goal.</td>
</tr>
<tr>
<td>( v_c )</td>
<td>The center mass vector caused by the collection of agents inside the visual range.</td>
</tr>
<tr>
<td>( v_{mv} )</td>
<td>The matching of the collection of agent’s velocity inside the visual range.</td>
</tr>
<tr>
<td>( v_a )</td>
<td>The Collision Avoidance vector caused by the collection of agents inside the comfort range.</td>
</tr>
<tr>
<td>( v_{vr} )</td>
<td>The visual terrain vector caused by the collection of terrain inside the visual range of the agent.</td>
</tr>
<tr>
<td>( v_j )</td>
<td>The agent’s current velocity.</td>
</tr>
<tr>
<td>( v_{max} )</td>
<td>This is the maximum velocity an agent can attain (default = 2.4).</td>
</tr>
<tr>
<td><strong>Comfort range</strong></td>
<td>The range around the agent will feel uncomfortable about other agents. (default = 3)</td>
</tr>
<tr>
<td><strong>Visual range</strong></td>
<td>The range an agent can see fellow agents and hazards; mountains, swamps, goals and map boundary. (default = 10)</td>
</tr>
</tbody>
</table>
\[ v_j = v_j \frac{2v_{\text{max}} k_j |v_j|}{1 + \frac{1}{e^{|v_j| v_{\text{max}}}}} \]  \hspace{1cm} (3.9)

Equation 3.10 updates the position of the agent on the map and checks for collisions. If a fellow particle is already located in the updated grid square a collision is recorded and the simulation’s algorithm sets the Y component of the vector to 0. If the X component is also a collision the agent’s vector is set to 0 and no movement occurs.

\[ x_j = x_j + v_j \]  \hspace{1cm} (3.10)

Table 3.1 summarizes the variables used in the previous equations.
CHAPTER IV
EXPERIMENTAL DESIGN

In this chapter we describe the experiments that we conducted to establish reasonable ranges of weight values and also describe methods used in experiments comparing homogeneous and heterogeneous populations. Finally we describe the methods for comparison of performance of homogeneous and heterogeneous swarms. In our experiments we have varied the weights of the different attributes rather than the attribute values themselves. Thus we are actually varying the importance of each attribute in determining the swarm behavior.

In order to evaluate the influence of heterogeneity on the performance of a swarm, a set of experiments were conducted using the initial attribute values of Winder and Reggia [21] to find reasonable ranges of weights for experimental parameters (attribute weights). Once reasonable ranges had been established for attribute weights, trials were conducted to compare the performance of homogeneous swarms and heterogeneous swarms for two maps in order to determine how to conduct these experiments. In homogenous swarms, all particles have identical weights for all attributes. In our experiments with heterogeneous swarms, we used homogeneous weights for all attributes except one. In heterogeneous swarms, the weights of one attribute are assigned values
from a Gaussian distribution with a mean that is the same as the reference homogeneous swarm.

4.1 Preliminary Experiments

The preliminary trials used the parameter settings of Winder and Reggia [21] non-memory experiments in order to establish a baseline for further experiments. There were two main differences in our environment and that used by Winder and Reggia [21]. The first is that their agents did not have any volume and there was no limit to the number of agents that could occupy one grid space whereas our agents have a volume of 1 grid square and therefore two agents cannot occupy the same grid space. Therefore, the resolution of our grid was increased by a factor of ten over that used by Winder and Reggia. The second difference is the vision method used to select the best path. The Winder and Reggia vision method used the unit vector weight of the closest edge in a grid square. In our implementation the agent computes a vector after looking at each grid square in its visual range and, taking obstacles into account, calculates the direction that allows the fastest velocity through the terrain.

4.1.1 Validation Experiment Results

Table 4.1 shows the weights and other experimental parameters used for the validation experiments comparing our results to those obtained by Winder and Reggia. These experiments varied Center Mass weight $w_c$, Collision Avoidance weight $w_a$ and Matched Velocity weight $w_{mv}$ over the same range reported by Winder and Reggia [21]. The results of our experiments are shown in Figure 4.1 and those of Winder and Reggia
are shown in Figure 4.2. The maximum value shown in Figure 4.1 is limited to 7000 to make it easier to view differences. The trials that reach 7000 on the chart actually encountered stalemate at 60,000 steps. The results of our experiments and those of Winder and Reggia are similar although not identical.

Table 4.1  Attribute Weights for Preliminary Experiments

<table>
<thead>
<tr>
<th>Test Cases</th>
<th>$w_c$</th>
<th>$w_{mv}$</th>
<th>$w_a$</th>
<th>Trials</th>
<th>Max # Steps</th>
<th>Maps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.04</td>
<td>0.2</td>
<td>0.40</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
<tr>
<td>2</td>
<td>0.04</td>
<td>0.2</td>
<td>0.80</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
<tr>
<td>3</td>
<td>0.20</td>
<td>0.2</td>
<td>0.40</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
<tr>
<td>4</td>
<td>0.04</td>
<td>0.4</td>
<td>0.40</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
<tr>
<td>5</td>
<td>0.40</td>
<td>0.4</td>
<td>0.40</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
<tr>
<td>6</td>
<td>0.76</td>
<td>0.4</td>
<td>0.04</td>
<td>20</td>
<td>60000</td>
<td>1-6</td>
</tr>
</tbody>
</table>

Figure 4.1  Validation experiments where the weight values used where the legend gives the values used for $<w_c, w_{mv}, w_a>$. 32
Figure 4.2  Winder and Reggia’s [21] results where the legend give the values used for $<w_c, w_{mv}, w_{av}>$.

These validation experiments gave a better understanding of the maps and how the weight values affects results. Maps 1 and 2 give almost no stalemate and Maps 3 and 4 give the most. Maps 5 and 6 take longer to complete because there are more goal points, but generally do not have stalemate.

4.1.2 Swarm Initialization

We experimented with two different methods to initialize the swarms on the maps. We compared the results when 1) the swarm was placed at a random position on the map with all particles within visual range of one another and 2) the swarm was placed in random grid spaces within visual range of the first goal. The random starting location did not appear to provide additional information and caused the swarm to take longer to reach the starting goal location and caused greater variation in the number of steps required to reach the goal. The wider variation made comparisons much more difficult.
and the longer running times limit the number of experiments that can be run. Therefore, we decided to use the second alternative of initializing the swarm within visual range of the first goal for subsequent experiments.

4.1.3 Approach for Establishing Reasonable Weight Ranges

A set of experiments was conducted to find reasonable ranges for the attribute weights. These experiments were conducted by varying the weight of the Center of Mass weight attribute for both homogenous and heterogeneous swarms. A weight value was considered reasonable if it met two criteria:

1. Swarms were able to complete all goals in a map.
2. Particles interactions occur.

The weights reported by Winder and Reggia [21] were used as a starting point for these experiments. One weight was varied at a time in estimated appropriate increments. The weights that are the focus of this work are $w_c$, $w_{mv}$ and $w_a$. The Center of Mass weight $w_c$ controls how the swarm stays grouped during movement where higher values imply stronger grouping. Center of Mass weights in the range 0.35 to 0.44 were acceptable for all maps. If the Center of Mass weight goes too high (e.g. over 0.5) then the swarm has so many collisions that it cannot make progress toward a goal (this condition is referred to as stalemate). Figure 4.3 illustrates how the swarm forms solid trail and eventually reaches a stalemate in the mountain terrain when the Center of Mass weight is very high.
The weight range for Collision Avoidance \( (w_a) \) and Velocity Matching \( (w_{mv}) \) also had acceptable ranges of 0.20 to 0.84 for all maps. Lower swarm cohesion causes the particles to separate and to be unable to coordinate on goal search. Higher swarm cohesion causes the particles to collide excessively and slow the goal search. When swarm cohesion is excessively high (weight above 1.0) the swarm reached stalemate as every particle is colliding and unable to perform goal searching.

A simulation terminates when a swarm completes all goals or after a predetermined maximum number of steps. In our initial experiments we used 60,000 steps as the upper limit.
4.1.4 Preliminary Heterogeneous Versus Homogeneous Experiment

The validation experiment had shown that swarms were most sensitive to changes in the Center of Mass weight and therefore we used that weight for our initial comparisons of heterogeneous and homogeneous swarms. Maps 2 and 4 were selected for this experiment because Map 2 was the fastest to complete and because Map 4 showed a high degree of variation in performance results with changes in weights. In our preliminary experiment with homogeneous swarms, we kept the Collision Avoidance (0.2) and Matched Velocity (0.4) weights constant and varied the Center of Mass Weight using the following values: 0.04, 0.13, 0.22, 0.31 and 0.40. We plotted the average number of steps to completion for 5 trials for each Center of Mass weight value. These values were plotted and a 2nd order polynomial was fit the line. The minimum value (number of steps) of this line was visually determined to occur at 0.18 and this value was selected as the mean for our subsequent experiment comparing homogeneous and heterogeneous swarms for Map 2 and Map 4. A Gaussian distribution with both 0.1 and 0.2 about the mean of 0.18 were tested. Figure 4.4 shows the results of this experiment. In general, there was little difference in the results with the two Gaussian distributions. We have chosen to use 0.1 for the standard deviation for subsequent experiments.
Figure 4.4 Performance of the swarms with different Center of Mass weight values on Map 2 and Map 4. Homogenous swarms with $w_c$ values from 0.04 to 0.40 are compared to heterogeneous swarms.

There were two interesting lessons learned from this experiment after examination of the detailed traces. First, the heterogeneous swarms in which some individuals had higher Center of Mass values than those that would typically cause stalemate were still able to complete successfully. Second, the number of collisions among the particles was about the same for the heterogeneous and homogeneous swarms.

As expected, we also observed that an increased number of collisions caused the swarm’s speed to decrease as shown graphically in Figure 4.5.
Figure 4.5  Comparison of the speed and the number of collision occurring in the swarm as the Center of Mass weight is varied (all swarms are homogeneous).

4.1.5  Heterogeneous Versus Homogenous Swarms for Near-Stalemate Weights

Our previous experiment had indicated that heterogeneous swarms with weight values near the margin for a particular situation might outperform homogenous swarms with the same mean weight value. Map 2 was selected for these experiments because the swarms complete this map in the fewest steps. We also reduced the maximum number of
steps to 20,000 to reduce the time required to complete this preliminary experiment. More trials (20) were conducted for each value to reduce the effect of outliers. In these experiments, the Center of Mass weight was varied for both homogeneous and heterogeneous trails.

The results shown in Figure 4.6 demonstrate that there was little difference in the performance of heterogeneous swarms for weights near the optimum but the heterogeneous swarms definitely have the advantage when the attribute is near the margin of the acceptable range for this map. This result led us to speculate that the diversity of individuals allows the swarm to solve problems in less time when they are in a situation where they typically encounter difficulty.

![Mean Steps to Completion](image)

Figure 4.6 Mean performance between homogenous and heterogeneous swarms when center mass is between 0.39 and 0.42 on Map 2.
4.2 Examples of Swarm Movement

We provide a few examples showing how swarms react on different maps. Figure 4.7 shows the simple Map 2 problem of avoiding a single hazard made of swamp and mountain. Agents will cross the swamp to pursue their goal.

Figure 4.7 Agent routing around the mountain terrain in Map 2 by exploring and following fellow agents around the hazard. Mountains are purple, swamp is gray, goal is red, agent path is green and white is open terrain.

Figure 4.8 illustrates a wide range of agent responses to varying hazards. When the agents have only open terrain a direct line to the goal is established, but more complicated behavior results when hazards are encountered.
Figure 4.8  Agent routing around multiple terrain hazards in Map 5. Mountains are purple, swamp is gray, goal is red, agent path is green and white is open terrain.

Figures 4.9 – 4.13 demonstrate a variety of cooperative behaviors exhibited by swarms as they pursue goals in different maps.
Figure 4.9  Map 5 hazards demonstrating an approximation of the u-shaped hazard researched by Wan and Chen [32].

Figure 4.10  Map 1 hazard of a large swamp, the agents veer around the swamp only touching the edges as it move to the next goal.
Figure 4.11  Map 3 hazards where the agents wandered around the top edge to find the right-side. Note how close the agents could of split with some going right and others going left.
Figure 4.12 Map 3 showing the completed path of the swarm about the hazards.
4.3 Experimental Design for Further Experiments

The maximum number of steps for our primary experiments was reduced to 40,000 because most simulation runs with Map 2 and Map 4 completed before 40,000 steps unless the trial was going to stalemate. This reduction in the maximum number of steps allowed reduced experiment simulation time and thus allowed for many more runs in a reasonable amount of time.

The goal weight $w_g$ and visual weight $w_{vr}$ were both set to 1.0 for these experiments. Although these two weights affect the performance of the system, they are not the focus of this current study. Table 4.2 outlines the simulation trials that were run to test our hypothesis. Note that the values used for each weight that was varied were
near the margin of acceptable values for Map 2 and Map 4. Ten simulations runs of no
more than 40,000 steps each were run for each weight value.

Table 4.2  Experiment Parameters

<table>
<thead>
<tr>
<th>Test Cases</th>
<th>Weight Variation</th>
<th>$w_c$</th>
<th>$w_{mv}$</th>
<th>$w_a$</th>
<th>Trials</th>
<th>Max Steps</th>
<th>Maps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Homogenous</td>
<td>0.39 to 0.44 in increments of 0.1</td>
<td>0.2</td>
<td>0.4</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
<tr>
<td>2</td>
<td>Gaussian 0.1</td>
<td>0.39 to 0.44 in increments of 0.1</td>
<td>0.2</td>
<td>0.4</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
<tr>
<td>3</td>
<td>Homogenous</td>
<td>0.3</td>
<td>0.75 to 0.84 in increments of 0.1</td>
<td>0.4</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
<tr>
<td>4</td>
<td>Gaussian 0.1</td>
<td>0.3</td>
<td>0.75 to 0.84 in increments of 0.1</td>
<td>0.4</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
<tr>
<td>5</td>
<td>Homogenous</td>
<td>0.3</td>
<td>0.2</td>
<td>0.75 to 0.84 in increments of 0.1</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
<tr>
<td>6</td>
<td>Gaussian 0.1</td>
<td>0.3</td>
<td>0.2</td>
<td>0.75 to 0.84 in increments of 0.1</td>
<td>10</td>
<td>40000</td>
<td>1-6</td>
</tr>
</tbody>
</table>

The overall simulation algorithm is outlined below:

- Initialize swarm engine, swarm, maps and user interface.
- Randomly place particles about the swarm starting location (first map goal) with all particles within visual range of the first goal.
- Randomly assign the initial direction for each particle.
• Start swarm to first goal
• Loop on each particle
• Calculate weights and vectors
• Check for collisions
• Update particle vectors and map position
• If 90% of particles are at map goal then go to update to the next map goal.
• Check if 40,000 loops has been reached (exit if true)
• Check if all map goals has been reached(exit if true)
• End loop
• Summarize all results and write out the results.

4.4 Simulation Parameters

Before starting a simulation, variables must be initialized to define a number of parameters for the specific simulation. These include the map to be used, goal locations, the step limit, and the attribute weights for each particle. In order to facilitate comparisons between runs, the particles of the swarm were always placed in close proximity to the first goal. This was achieved by using the initial goal as the center of a circle whose radius was the visual range of the particles. The particles were then placed in random locations within this circle. Initial velocity vectors were randomly set using a uniform distribution in the range [0 ... maximum velocity] where the maximum velocity is 2.5. The other preset parameters for maximum speed, mountain terrain, swamp terrain, visual range and comfort range constants are adapted from those used by Winder and
Reggia [21] and are scaled appropriately to account for the increased resolution of our maps. These parameters are listed in Table 3.1.

Each swarm contained 10 particles in all of our experiments. Homogenous swarm particles have the same weight values for all particles. Heterogeneous swarm particles are identical to homogeneous swarm particles except that the weight of a single parameter was varied. Weights for the particles of heterogeneous swarms were generated using a random Gaussian distribution with a specified mean and a standard deviation of 0.1. When homogeneous and heterogeneous swarms are compared, the mean of the weights for the heterogeneous swarm is the same as the constant weight used for the corresponding homogeneous population.

The simulation system was run on an MS Windows PC, 1.6 GHz and 512 MB RAM using the Swarm toolkit [31]. The Swarm toolkit was run using Java 1.5.0_03 launched by a DOS batch file. Debugging of the initial experiments was assisted by code walking by using the NetBeans 4.1 debugger.

4.5 Testing for Statistical Significance of Performance Differences

The hypothesis about performance will be evaluated primarily though the number of steps a simulation requires before it completes. A simulation is considered complete when either 90% of the particles reach all goals or 40,000 simulation steps have been completed. The primary metric for evaluation is the difference between the means of steps to completion for comparable heterogeneous and homogeneous swarms. Evaluation of differences between performance of heterogeneous and homogeneous populations was done using Student’s T-test to compare the means of trial results to determine if the
performance is significantly different. That is we are testing the hypothesis that the difference in the mean values of our experiments is attributable to something other than chance—that it is attributable to using heterogeneous swarms versus homogeneous swarms. Several additional characteristics of the swarms were also captured for secondary analysis of results to determine if any patterns could be observed in the group dynamic. Other measurements were: Number of collisions, number of particles in visual range, number of steps in open, swamp, and mountain terrain, particle speed, and rank of the particles in the order of arrival at the goal.
CHAPTER V
EXPERIMENTAL RESULTS AND ANALYSIS

The goal of this research is to investigate the differences, if any, in the performance capability of swarms of heterogeneous agents and swarms of homogeneous agents. Preliminary experiments indicated that when attribute weights of the entire swarm are near the optimum (all have the same capabilities and these are the best values for the particular environment) there is little difference in the performance of heterogeneous and homogeneous swarms. However, when the attribute weights are not optimal for the situation, the heterogeneous swarms exhibit superior performance. The hypothesis of our research is the following:

Compared to swarms with homogeneous attribute weights, swarms with heterogeneous attribute weights can, when the mean weight is near the margin of acceptable values, 1) reach the goal in fewer steps and 2) solve some problems that homogeneous swarms cannot solve.

Experiments were conducted to confirm or refute this hypothesis and statistical analysis was done to determine the significance of the findings. Results are analyzed separately for each weight type (Center of Mass, Collision Avoidance, and Matched Velocity). This is followed by an overall evaluation of performance differences between homogenous and heterogeneous swarms.
5.1 Center of Mass Weight Results

Table 5.1 lists the performance results for the Center Mass Weight test cases. In these cases, other weights were maintained at their optimal value and the Center of Mass values near the limit of acceptable values were varied for both homogeneous and heterogeneous swarms. Performance is defined as the number of steps it takes for a swarm to complete the map goals or reach 40,000 steps (the cut-off for the simulation). The performance numbers are the averages for 10 simulation runs for each weight value and have been rounded to the nearest whole number. These results are shown graphically for each map in Figure 5.1.

<table>
<thead>
<tr>
<th>Weights</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.35</td>
<td>2037</td>
<td>1131</td>
<td>24934</td>
<td>3970</td>
<td>4834</td>
<td>4514</td>
</tr>
<tr>
<td>0.36</td>
<td>2031</td>
<td>1190</td>
<td>32505</td>
<td>6076</td>
<td>4894</td>
<td>7974</td>
</tr>
<tr>
<td>0.37</td>
<td>2033</td>
<td>1071</td>
<td>18077</td>
<td>6688</td>
<td>4807</td>
<td>4502</td>
</tr>
<tr>
<td>0.38</td>
<td>2041</td>
<td>5035</td>
<td>40000</td>
<td>6474</td>
<td>4893</td>
<td>11704</td>
</tr>
<tr>
<td>0.39</td>
<td>2038</td>
<td>1247</td>
<td>34705</td>
<td>9172</td>
<td>4807</td>
<td>4502</td>
</tr>
<tr>
<td>0.40</td>
<td>2044</td>
<td>9025</td>
<td>34324</td>
<td>4311</td>
<td>4873</td>
<td>4599</td>
</tr>
<tr>
<td>0.41</td>
<td>2064</td>
<td>5022</td>
<td>40000</td>
<td>11460</td>
<td>5023</td>
<td>8172</td>
</tr>
<tr>
<td>0.42</td>
<td>2056</td>
<td>9273</td>
<td>40000</td>
<td>11293</td>
<td>4730</td>
<td>25775</td>
</tr>
<tr>
<td>0.43</td>
<td>2052</td>
<td>13235</td>
<td>37267</td>
<td>19290</td>
<td>5109</td>
<td>15192</td>
</tr>
<tr>
<td>0.44</td>
<td>2072</td>
<td>21074</td>
<td>40000</td>
<td>18284</td>
<td>5163</td>
<td>18840</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weights</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.35</td>
<td>2073</td>
<td>1415</td>
<td>40000</td>
<td>6355</td>
<td>4934</td>
<td>4645</td>
</tr>
<tr>
<td>0.36</td>
<td>2087</td>
<td>5357</td>
<td>40000</td>
<td>7175</td>
<td>4791</td>
<td>8299</td>
</tr>
<tr>
<td>0.37</td>
<td>2104</td>
<td>1782</td>
<td>40000</td>
<td>9227</td>
<td>4957</td>
<td>11858</td>
</tr>
<tr>
<td>0.38</td>
<td>2114</td>
<td>9488</td>
<td>40000</td>
<td>11686</td>
<td>5066</td>
<td>15443</td>
</tr>
<tr>
<td>0.39</td>
<td>2134</td>
<td>17822</td>
<td>40000</td>
<td>14678</td>
<td>5141</td>
<td>25844</td>
</tr>
<tr>
<td>0.40</td>
<td>2145</td>
<td>14102</td>
<td>40000</td>
<td>12899</td>
<td>4885</td>
<td>33026</td>
</tr>
<tr>
<td>0.41</td>
<td>2151</td>
<td>28694</td>
<td>40000</td>
<td>15594</td>
<td>5142</td>
<td>25964</td>
</tr>
<tr>
<td>0.42</td>
<td>2170</td>
<td>40000</td>
<td>40000</td>
<td>18804</td>
<td>5104</td>
<td>40000</td>
</tr>
<tr>
<td>0.43</td>
<td>2188</td>
<td>40000</td>
<td>40000</td>
<td>18142</td>
<td>5178</td>
<td>29417</td>
</tr>
</tbody>
</table>
Figure 5.1  Performance of swarms for varying values of the Center of Mass weight.

The results in Table 5.1 and Figure 5.1 show that swarms with heterogeneous weights for the Center of Mass weight usually perform better than homogenous swarms for almost every trial with attribute weights near the margin. The exception is Map 5 with a Center of Mass weight of 0.39. Examination of individual results show that one run on Map 5 of the heterogeneous swarm with weight 0.39 trial encountered stalemate, but all other runs on Map 5 completed. This single outlier caused the Map 5 heterogeneous mean to be excessively high.

Table 5.2 gives the p-values on Student’s t-test comparing the statistical significance of the difference in the means for the heterogeneous and homogeneous
experiments when varying Center of Mass weight. For this thesis, we are interested in a statistically significant difference at the 95% confidence level. That is, for p-values less than 0.05, we determine that there is a statistically significant difference in the two means.

Table 5.2  Student T-Test Results Comparing Homogeneous and Heterogeneous Swarms for Center of Mass Attribute Weight

<table>
<thead>
<tr>
<th>Center Mass Weight</th>
<th>T-Test</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.35</td>
<td>1.67E-06</td>
<td>0.857021</td>
<td>0.083472</td>
<td>*</td>
<td>4.38E-05</td>
<td>0.189726</td>
<td>0.100196</td>
</tr>
<tr>
<td>0.36</td>
<td>4.72E-08</td>
<td>0.379482</td>
<td>0.199771</td>
<td>*</td>
<td>0.784788</td>
<td>0.757322</td>
<td>0.169575</td>
</tr>
<tr>
<td>0.37</td>
<td>1.38E-11</td>
<td>0.289322</td>
<td>0.005275</td>
<td>*</td>
<td>0.870366</td>
<td>0.908077</td>
<td>0.686284</td>
</tr>
<tr>
<td>0.38</td>
<td>7.38E-05</td>
<td>0.42192</td>
<td>*</td>
<td>0.255456</td>
<td>0.515346</td>
<td>0.132469</td>
<td></td>
</tr>
<tr>
<td>0.39</td>
<td>5.01E-10</td>
<td>0.140537</td>
<td>0.343436</td>
<td>0.544052</td>
<td>0.373791</td>
<td>0.233811</td>
<td></td>
</tr>
<tr>
<td>0.40</td>
<td>2.24E-11</td>
<td>0.284786</td>
<td>0.343436</td>
<td>0.001761</td>
<td>0.078559</td>
<td>0.005201</td>
<td></td>
</tr>
<tr>
<td>0.41</td>
<td>8.45E-09</td>
<td>0.205239</td>
<td>*</td>
<td>0.623986</td>
<td>0.290464</td>
<td>0.076427</td>
<td></td>
</tr>
<tr>
<td>0.42</td>
<td>0.000129</td>
<td>0.021658</td>
<td>*</td>
<td>0.108141</td>
<td>0.008117</td>
<td>0.002622</td>
<td></td>
</tr>
<tr>
<td>0.43</td>
<td>1.15E-12</td>
<td>0.001325</td>
<td>*</td>
<td>0.809412</td>
<td>0.911402</td>
<td>0.020488</td>
<td></td>
</tr>
<tr>
<td>0.44</td>
<td>5.7E-05</td>
<td>0.014995</td>
<td>*</td>
<td>0.943162</td>
<td>0.861806</td>
<td>0.014427</td>
<td></td>
</tr>
</tbody>
</table>

* Identical means for both homogeneous and heterogeneous experiments

The t-test results presented in Table 5.2 show that there heterogeneity does make a difference on map 1 across all values of the Center of Mass weight. For other maps this consistency is not shown, but for all values there is at least one weight value for which heterogeneity does make a significant difference.

The results of this experiment support the hypothesis that heterogeneous swarms will outperform homogenous swarms as the weights near the margin of acceptable values for a particular situation. We also found that for Map 3, the heterogeneous swarms were able to reach the goal within 40,000 steps when the homogeneous swarms could not.
5.2 Collision Avoidance Weight Results

The Collision Avoidance weight ranged from 0.75 to 0.84 because, based on our experiments with Map 2 and Map 4, we determined that these values were likely to be at the margin of those causing stalemate. Results from this experiment are shown in Table 5.3 and Figure 5.2. The same problems occurred with stalemate on Map 3 as in the previous experiments. The surprise is that a few of the trials did not stalemate for the heterogeneous Collision Avoidance weight test cases at 0.75 and 0.78. In general, the performance of the swarm was not sensitive to changes in the Collision Avoidance weight. We hypothesize that this lack of sensitivity is because the equation for the Collision Avoidance vector calculation only uses neighbors that are very near. This means a range of only 2.7 grid squares is used as compared to the Center of Mass vector that takes up to 10 neighboring grid squares into account.

A comparison of the performance of heterogeneous and homogenous swarms when the Collision Avoidance weight was varied show that there is a statistically significant difference in the mean performance values of homogeneous and heterogeneous populations for some weight values on some maps. Specifically, on map 1 with weight values of 0.79 and 0.83, map 2 with a weight value of 0.79, map 4 with a weight value of 0.84, and map 6 with weight values of 0.79 and 0.84. In all but one of these cases where statistically different results were observed, the homogeneous swarm outperformed the heterogeneous swarm. We conclude the heterogeneous values for the Collision Avoidance weight attribute are not helpful, and in fact, may be detrimental.
Table 5.3  Collision Avoidance Weight performance for both heterogonous and homogenous swarms.

<table>
<thead>
<tr>
<th>Weights</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.75</td>
<td>2075.98</td>
<td>1433.96</td>
<td>37861.1</td>
<td>3330.38</td>
<td>5330.89</td>
<td>4681.44</td>
</tr>
<tr>
<td>0.76</td>
<td>2081.47</td>
<td>1607.14</td>
<td>40000</td>
<td>3232.22</td>
<td>5136.57</td>
<td>4683.83</td>
</tr>
<tr>
<td>0.77</td>
<td>2081.27</td>
<td>1608.49</td>
<td>40000</td>
<td>3501.7</td>
<td>5169.64</td>
<td>4757.37</td>
</tr>
<tr>
<td>0.78</td>
<td>2102.73</td>
<td>1903.69</td>
<td>38088.3</td>
<td>3649.39</td>
<td>5275.2</td>
<td>4763.67</td>
</tr>
<tr>
<td>0.79</td>
<td>2090.08</td>
<td>1414.23</td>
<td>40000</td>
<td>3604.65</td>
<td>5190.86</td>
<td>4800.46</td>
</tr>
<tr>
<td>0.80</td>
<td>2086.65</td>
<td>1456.77</td>
<td>40000</td>
<td>3633.44</td>
<td>5486.18</td>
<td>4756.45</td>
</tr>
<tr>
<td>Mean</td>
<td>2093.01</td>
<td>1597.57</td>
<td>39884.8</td>
<td>3629.86</td>
<td>5297.23</td>
<td>4763.67</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weights</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.75</td>
<td>2077.98</td>
<td>1483.69</td>
<td>40000</td>
<td>2798.83</td>
<td>5322.34</td>
<td>4789.46</td>
</tr>
<tr>
<td>0.76</td>
<td>2079.15</td>
<td>1456.74</td>
<td>40000</td>
<td>3192.15</td>
<td>5336.56</td>
<td>4619.53</td>
</tr>
<tr>
<td>0.77</td>
<td>2078.22</td>
<td>1653.6</td>
<td>40000</td>
<td>3164.41</td>
<td>5467.33</td>
<td>4621.56</td>
</tr>
<tr>
<td>0.78</td>
<td>2080.84</td>
<td>1448.87</td>
<td>40000</td>
<td>3456.68</td>
<td>5309.82</td>
<td>4633.69</td>
</tr>
<tr>
<td>0.79</td>
<td>2040.57</td>
<td>933.33</td>
<td>40000</td>
<td>3269.5</td>
<td>5245.73</td>
<td>4438.64</td>
</tr>
<tr>
<td>Mean</td>
<td>2079.21</td>
<td>1276.51</td>
<td>40000</td>
<td>3269.5</td>
<td>5245.73</td>
<td>4438.64</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weights</th>
<th>Map 1</th>
<th>Map 2</th>
<th>Map 3</th>
<th>Map 4</th>
<th>Map 5</th>
<th>Map 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.80</td>
<td>2088.31</td>
<td>1743.7</td>
<td>40000</td>
<td>3375.33</td>
<td>5509.81</td>
<td>4901.84</td>
</tr>
<tr>
<td>0.81</td>
<td>2086.71</td>
<td>2174.38</td>
<td>40000</td>
<td>3091.84</td>
<td>5411.69</td>
<td>4755.58</td>
</tr>
<tr>
<td>0.82</td>
<td>2089.68</td>
<td>2107.38</td>
<td>40000</td>
<td>3700.43</td>
<td>5414.84</td>
<td>4808.8</td>
</tr>
<tr>
<td>0.83</td>
<td>2088.13</td>
<td>2226.77</td>
<td>40000</td>
<td>4037.37</td>
<td>5384.08</td>
<td>4873.56</td>
</tr>
<tr>
<td>0.84</td>
<td>2092.59</td>
<td>2080.95</td>
<td>40000</td>
<td>3697.15</td>
<td>5491.39</td>
<td>4525.35</td>
</tr>
<tr>
<td>Mean</td>
<td>2090.14</td>
<td>2091.6</td>
<td>40000</td>
<td>3697.15</td>
<td>5491.39</td>
<td>4525.35</td>
</tr>
</tbody>
</table>
Figure 5.2 Collision Avoidance Weight performance for both heterogeneous and homogenous swarms.

5.3 Matched Velocity Weight Results

We selected the range of values for the Matched Velocity weight of 0.75 – 0.84 because, based on experiments with Map 2 and Map 4, we thought these values were near the margin of acceptability. For this range of values, there was very little difference in the performance of heterogeneous and homogeneous swarms. However, as the results in Table 5.4 and Figure 5.3 show, with these high values for the Matched Velocity, the swarms were able to complete goals successfully in a relatively small number of steps when the swarm had consistently encountered stalemate on this map with the previously
used lower value weights. When examining the statistically significant differences in the means we find that in these set of experiments heterogeneity makes a differences on maps 2, 3, and 4 for weight values of 0.84, and on map 4 with a weight value of 0.81. In these cases with very high Matched Velocity values, the homogeneous swarms exhibited the better performance. We have completed an additional experiment based on these results to compare heterogeneous and homogenous swarms over a larger range of values for this attribute weight for Map 3.

Table 5.4 Matched Velocity Weight performance for both heterogenous and homogenous swarms.
Figure 5.3 Matched Velocity weight performance for both heterogeneous and homogenous swarms

5.4 Summary

The results of these experiments have shown that when the values of weights are at the limits of those optimal for a specific situation, swarms with heterogeneous values for the Center of Mass weight often outperform homogeneous swarms. The weight of the Collision Avoidance parameter had little effect on the performance of the swarms and few differences were seen in the performance of heterogeneous and homogeneous swarms. Higher weights for the Matched Velocity parameter, to our surprise, allowed the swarms to solve maps that it could not solve with lower weights. There were few
differences seen for heterogeneous and homogenous swarms when the Matched Velocity parameter had a higher weight. For intermediate weights, heterogeneous swarms outperformed homogeneous swarms and heterogeneous swarms could solve maps that the homogeneous swarms could not solve.
CHAPTER VI
CONCLUSION

The focus of this thesis has been a careful measurement of the differences in performance of heterogeneous and homogeneous swarms. Although a number of studies have discussed the advantages of diversity in swarms, there have been, to our knowledge, no previous studies that carefully addressed this problem. Based on preliminary experiments we had formulated the hypothesis that swarms with heterogeneous weights for attributes, when compared to homogeneous swarms, would, when the mean weight is near the margin of acceptable values, 1) reach the goal in fewer steps and 2) solve some problems that homogeneous swarms cannot solve.

We conducted a set of experiments in which weights for a single attribute were allowed to take on heterogeneous values based on a Gaussian distribution while others were held constant. Our results demonstrate that our hypothesis holds for heterogeneous weights for the Center of Mass attribute weight. Swarms heterogeneous in this attribute weight reached the goals in fewer steps than their homogeneous counterparts and could solve some maps that the homogeneous swarms could not. We speculate that this is due to the fact that, in a heterogeneous swarm, some of the particles will be able to pull the rest of the swarm toward the goal even if those individual particles are poorly suited to solve the goal. The weight of the Collision Avoidance attribute seemed to have little
effect on performance and there were few differences in the heterogeneous and homogeneous swarms. The high values of Matched Velocity that we investigated had a surprisingly good performance particularly for the most difficult map. In subsequent experiments, we determined that, for this map, heterogeneous swarms had a definite advantage over homogeneous ones. We speculate that the higher weight of Matched Velocity allows the swarm to remain cohesive and steer around the large difficult obstacles in Map 3.

The major contribution of this work is the clear demonstration of the advantage of diversity within a swarm. The advantage of diversity is most clearly seen when the swarm encounters a situation for which it is not particularly well-suited. In these situations, one of the particles that is more well-suited to the environment appears to be able to lead the others to a solution even though some of the particles have parameter values that would typically result in stalemate. This result could have important implications when swarms are used to solve real world problems where the situations they will encounter are unpredictable.

6.1 Future Work

One of the greatest limitations of our experiments with simulations is the length of time required to run each trial. If greater processing power were available, it would be possible to conduct more experiments and to experiment with diversity in the other parameters. In addition, our experiments were limited to diversity in a single parameter weight. Further experiments in which heterogeneous weights are used for multiple weights simultaneously could also reveal interesting differences.
Other extensions of our work could include looking at methods for characterizing terrains over which these swarms interact. As seen for the set of experiments dealing with the Center of Mass weight, one map clearly indicated the advantage of using heterogeneous populations while other maps did not reveal quite as dramatic a difference. It would be useful to design a protocol for methodically examining the effect of various topologies on the performance of homogeneous and heterogeneous swarms.
REFERENCES


