IMPROVING OPERATOR RECOGNITION AND PREDICTION OF EMERGENT SWARM BEHAVIORS

by

Phillip M. Walker

B.A. in Economics & Cognitive Science, College of William and Mary, U.S.A., 2010

M.S. in Intelligent Systems, University of Pittsburgh, U.S.A., 2012

Submitted to the Graduate Faculty of
the Kenneth P. Dietrich School of Arts and Sciences in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

DIETRICH SCHOOL OF ARTS AND SCIENCES

This dissertation was presented

by

Phillip M. Walker

It was defended on

March 2, 2017

and approved by

Michael Lewis, School of Information Sciences, University of Pittsburgh

Katia Sycara, Robotics Institute, Carnegie Mellon University.

Christian Schunn, Learning Research and Development Center, University of Pittsburgh

Stephen Hirtle, School of Information Science, University of Pittsburgh

Dissertation Director: Michael Lewis, School of Information Sciences, University of Pittsburgh

ABSTRACT

IMPROVING OPERATOR RECOGNITION AND PREDICTION OF EMERGENT SWARM BEHAVIORS

Phillip M. Walker, PhD

University of Pittsburgh, 2017

Robot swarms are typically defined as large teams of coordinating robots that interact with each other on a local scale. The control laws that dictate these interactions are often designed to produce emergent global behaviors useful for robot teams, such as aggregating at a single location or moving between locations as a group. These behaviors are called emergent because they arise from the local rules governing each robot as they interact with neighbors and the environment. No single robot is aware of the global behavior yet they all take part in it, which allows for a robustness that is difficult to achieve with explicitly-defined global plans. Now that hardware and algorithms for swarms have progressed enough to allow for their use outside the laboratory, new research is focused on how operators can control them. Recent work has introduced new paradigms for imparting an operator's intent on the swarm, yet little work has focused on how to better visualize the swarm to improve operator prediction and control of swarm states. The goal of this dissertation is to investigate how to present the limited data from a swarm to an operator so as to maximize their understanding of the current behavior and swarm state in general. This dissertation develops—through user studies—new methods of displaying the state of a swarm that improve a user's ability to recognize, predict, and control emergent behaviors. The general conclusion is that how summary information about the swarm is displayed has a significant impact on the ability of users to interact with the swarm, and that future work should focus on the properties unique to swarms when developing visualizations for human-swarm interaction tasks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

PREFACE

I would like to begin by expressing my thanks to my advisor and mentor during my time at the University of Pittsburgh, Dr. Michael Lewis, for his guidance, support, and encouragement for my work and career. He set a perfect balance of allowing me freedom to pursue my own interests while still keeping me honest. I will truly appreciate his mentorship.

I would also like to extend my gratitude to the other members of my committee: Dr. Katia Sycara, Dr. Christian Schunn, and Dr. Stephen Hirtle, for their guidance and advice in performing this dissertation work. Similarly, to my other colleagues at the University of Pittsburgh and Carnegie Mellon University, without whom my time here would have been much less enjoyable: Shih-Yi Chien, Andreas Kolling, Steven Nunnally, William Garrard, Huadong Wang, Nilanjan Chakraborty, Sasanka Nagavalli, and many others.

Finally, I would like to give special thanks to my family, close friends, and partner in life, Marisa Sprowls. Without their love and encouragement this journey would not have been possible. Their faith in me kept my spirit alive during the last few years, and I am forever grateful for their confidence in me.

1.0 INTRODUCTION

Robot swarms consist of multiple autonomous or semi-autonomous robots that coordinate via local control laws. These laws are based on the robot's current state and surrounding environment [11]. Key advantages of robotic swarms include robustness to failure of individual robots and scalability [108, 7], both of which are due to the *simple* and *distributed* nature of their coordination. Swarms are distinct from multi-robot systems in that the latter have explicitly represented goals, form and execute both individual and group plans, have different capabilities, and can assume different roles [31, 97, 70]. Such robots could act independently without coordinating, e.g., multiple robots searching a different area for victims in a search and rescue scenario. Conversely, they could also cooperate as a team in which all members work towards known shared goals, or coalitions in which members are self-interested. Swarms, on the other hand, involve coordination between robots that relies on autonomous distributed algorithms and information processing. Individual robots in the swarm are therefore not typically directly addressable by a central user. Because of this, global behaviors cannot be explicitly stated and instead must emerge from local interactions. The individual members of a swarm likely could not act independently in any successful manner with respect to the main goal without significant redesign.

The origins of swarm robotics can be found in early work investigating collectives of swarms in nature, such as the work by Iain Couzin [23], which in turn draws inspiration from the established field of self-organization in biological systems (see [15, 73] for examples). However, in the past decade swarm robotics has become an engineering discipline in its own right [9, 4, 11, 127], with numerous researchers focused on improving the hardware and algorithms that support swarms. Swarm robotics promises a wide range of benefits and applications, from improving coverage during monitoring and reconnaissance missions [19], to improving tracking and search and rescue of multiple targets [122]. Even NASA has investigated the use of robot swarms in space for future missions [131].

Despite the benefits, much of the research claiming to use robot swarms are actually still multi-robot systems, in that the swarm is controlled through a centralized planner, or is not scalable to larger number of robots. The main reason for this is likely because it is far from clear how human operators can influence the behavior of swarms after deployment, while still maintaining the "swarmish" characteristics of the collective. Consequently, true swarms of robots are currently limited to laboratory settings only, with rigid, pre-defined behaviors and numbers often less than 100. Even the largest swarms, such as the recent kilobots developed at Harvard [106], have limited capabilities and provide few options for human intervention after deployment. Because of these difficulties, visualization and control of robot swarms by humans, hereafter referred to as human-swarm interaction (HSI), has been largely ignored until recently, with researchers instead focusing on improving autonomous swarm algorithms, communication abilities, and robot hardware in general.

One part of the difficulty in HSI comes from the fact that individual robots are typically not addressable, and global goal behaviors cannot be directly specified. Therefore, an operator must instead influence the interactions between robots indirectly, in a manner which moves the entire swarm along a trajectory to the intended goal. This can be achieved through a variety of paradigms, such as behavior selection, parameter tuning, environmental influence—such as the use of virtual pheromones—and using leaders as intermediaries. Although these paradigms can aid in proper *control* of the swarm, the other side of the control loop—*visualization*—is even less explored. Due to the distributed nature of the swarm and restrictions in the communication channels between the operator and numerous robots, complete information about its full global state often cannot be directly accessed. Instead, operators must make due with summary information and use indirect or incomplete observation to infer the global state. This inherently makes knowing the proper control input difficult, and thus improving the visualization techniques used to display a swarm to an operator is key to advancing the field of HSI overall.

The final element to be considered in HSI is that of trust. There is a significant body of work investigating human trust in automated systems, including robots specifically, but it remains to be seen if swarms are viewed and trusted in the same manner. The element of trust that will be highlighted and investigated in this thesis is that of transparency (or "process" in [67] and "integrity" in [76]), which is simply how much information about the automation is provided to the human during the automation's operation. Previous work has shown that a high level of transparency increases trust over lower levels of transparency [10, 140], but can also decrease reliance on the automation [68]. However, because a high degree of transparency for swarms can be both difficult to achieve (due to the large number of robots and thus large amount of information to be communicated) and overwhelming for the operator, designers of visualizations for HSI systems need to take special care to leverage the amount of bandwidth—both in the communication channel and in the operator's cognitive abilities—with the state information available to give operators the best picture of the swarm's current state for the current task.

To summarize, the difficulties in HSI are threefold, and capture the main aspects of the control loop that are unique to interacting with swarms as opposed to single robots or multi-agent systems. They can be summarized in three general questions, which will be used as the general inspiration for this thesis:

1. *Visualization*: How can an interface display incomplete information returned from the swarm in a manner that gives the operator the best awareness of the current state, and how does the current behavior affect how the swarm should be displayed?
2. *Prediction*: How does this visualization affect the operator's ability to make accurate predictions about the future state and the effect of their commands?
3. *Trust*: How does the behavior and visualization of the swarm impact the user's trust in the swarm, and does higher trust correlate with higher performance?

These questions briefly summarize a small amount of the significant research still needed before human-controlled swarms can operate effectively in the real world. As proper visualization and prediction is a necessary prerequisite for proper control, this thesis will focus primarily on the swarm-to-human visualization channel. These questions will guide the discussion and work presented in this thesis, and will be more specifically stated in Section 1.2.

1.1 MOTIVATION

The motivation for this dissertation comes from the intersection of three observations about swarms, and how human operators can and cannot interact with them.

First is the fact that, in many real-world cases, communication bandwidth and latency, compounded with the limited hardware available on each robot, are such that perfect information about every member of the swarm cannot be updated in real time. Even if full information can be displayed, it might be overwhelming for an operator to keep track of the entire swarm and understand the global state, especially for very large swarms. Furthermore, due to the decentralized nature of the swarm, presenting global information to a user at a centralized interface requires special care to balance the robustness that decentralization provides with the need for a central operator to understand what is going on. This balance can be addressed in multiple ways—from having leaders selected in a distributed manner aggregate information about the swarm for a centralized interface to put together, to having the human operate amongst the swarm itself. In all cases, the information available needs to be presented to the human in an intelligent way, so that the operator can focus on determining the next proper course of action rather than deciphering the current state of the swarm.

The second observation is that human exposure to swarms is every-day life is limited, so researchers have little idea both how operators view swarms and how they can learn to predict properties about the emergent behavior of a swarm, hence the question of prediction. This means that research is required to determine what exactly displaying information in an intelligent manner means, which requires research into how humans view collective agents and emergent systems. The literature on human perception of biological motion and common fate can be of particular interest here, as swarms are often seen as moving collectives, and many artificial swarms take inspiration from biological swarms, such as bird flocks and schools of fish—collectives that humans are more familiar with. The research in these related fields will be reviewed in later sections and incorporated into the work presented in this thesis.

The third observation is that control of swarms is not as intuitive as control of a single robot. For instance, previous work has introduced the concept of neglect benevolence: the idea that a swarm may need time to evolve or move toward consensus between subsequent inputs [138, 87]. While subsequent research showed that participants can learn to adapt to this phenomenon over time [86], the results suggest that proper visualization and prediction of the swarm state is necessary for proper control, as operators cannot immediately know when input should be given without assistance from the display.

Not coincidentally, these questions are all centered around the problem of visualization. When faced with incomplete information from a distributed collective of agents, designers of an HSI system must take special care in displaying this information to the user in a manner that is both easily digestible, and more informative than the raw data. Furthermore, we must develop simple tasks and metrics to evaluate different methods of displaying this information for different behaviors, so that new systems can have a basis to start from and be easily compared to existing ones. Finally, we must determine the role trust plays in HSI, which in turn will dictate how visualizations should be designed: the visualization should achieve the right level of transparency to properly calibrate operator trust in the system.

There is significant overlap between the three general questions presented in the introduction that can be taken advantage of when answering them. For instance, the questions of visualization and prediction are naturally intertwined. An intelligent visualization of available state information can improve operator prediction, and similarly better prediction allows operators to better understand future visualization of the swarm state. Therefore, the first goal of this thesis is to use the results of existing studies on visualizations for swarms (e.g. [138, 90]) and control input timing (e.g. [138, 86]) to develop user studies that improve our knowledge both about how swarm information can be visualized and how prediction improves as a result.

Similarly, the question of trust is naturally intertwined with the questions of visualization and prediction. We know from the large body of work on trust in automation that proper calibration of trust increases performance [67, 80], and because transparency (i.e. the visualization of the swarm) plays a role in the level of trust operators have in the system, it is necessary to consider trust when investigating visualizations for swarms. The field of trust in HSI is only beginning to be thoroughly researched however, so the work presented in this thesis on trust will likely raise even more questions than it answers.

1.2 RESEARCH QUESTIONS

In light of the above discussion, and before introducing the background of the concepts discussed so far when reviewing prior related work, I will explicitly state the research questions motivating this thesis. They will guide the formation of hypotheses for the studies presented herein, and are generally a good reference for the structure of the work proposed. The questions are as follows:

1. What variables or characteristics of a swarm are the most valuable to improving an operator's ability to understand and predict the swarm state?
2. Can a visualization that maximizes operators' abilities to understand and predict the swarm also improve their ability to give proper control the swarm?
3. Finally, can a visualization that achieves good results in accordance with the prior two questions also improve the operator's trust and confidence in the swarm, and is there a risk of a good visualization making operators *too* trusting?

Also note that these questions roughly correspond to each of the user studies presented in this thesis. The first question is addressed by the Recognition (Section 4.1) and Prediction (Section 4.2) studies, the second by the Control study (Section 4.3), and the third by the Trust study (Section 4.4). I will now present the overall structure of this thesis, and then review the relevant work in related fields needed to place this thesis into context.

1.3 THESIS STRUCTURE

The thesis will follow the questions posed in the previous section, and will begin with a discussion of related work in Chapter 2, including an introduction to biologically-inspired swarm behaviors (Section 2.1) and an overview of the limited amount of previous work in visualizing swarms and other collectives (Section 2.2). Also included in the literature review are discussions of other subfields relevant to this work, such as different control paradigms for swarms (Section 2.3), neglect benevolence in swarms (Section 2.4), biological motion and common fate (Section 2.5), and trust in automation (Section 2.6). After the literature review, Chapter 3 will introduce the emergent behaviors and visualizations used for the studies herein, as well as an overview of the simulator architecture used to make the studies possible. Chapter 4 presents the four user studies that make up the contribution of this thesis. The first study investigates human recognition of different swarm behaviors against background noise (Section 4.1), the second study investigates prediction of future swarm states when operating under these different behaviors (Section 4.2), the third study investigates control of swarms (Section 4.3), and the final study looks at whether improving the visualization of swarm data also improves operator trust in the swarm (Section 4.4). A discussion of these studies, along with a summary of this work's contribution and suggestions for future research, will be presented in Chapter 5.

2.0 RELATED WORK

In the following sections, I will review the relevant literature in swarm robotics and HSI necessary for this thesis, highlighting existing holes and demonstrating how my research questions will address them. I will also attempt to situate this work within the wider scientific community, so that it's results may be applicable outside HSI. This chapter will include first an overview of biologically-inspired swarms (Section 2.1). While there are numerous metaphors and models for swarm operation in addition to the bio-inspired model, the bio-inspired model will be used herein primarily because is one of the more common and well-researched models, and because it will be used as the primary metaphor for the experiments proposed. The metaphor used in the designing of swarm algorithms is also important for the visualization, as giving the operator an intuitive understanding of how members of a swarm interact with each other can aid in predicting future states and the effect of new commands.

Following the introduction of bio-inspired swarms, I will review existing literature in swarm state visualization and prediction in Section 2.2. Much of the work in visualization and prediction is originally intended to be research into control of swarms—with the visualization added as an afterthought—yet there are crucial lessons that can be learned from them in terms of how information should (and should not) be presented. This section will also present work on non-visual methods of communicating swarm state information to operators, such as haptic feedback. Following this discussion, I will introduce prior work on control of swarms in Section 2.3. This will allow the reader to understand the many paradigms for swarm control, including setting control law parameters and leader selection. Understanding leader selection is key to properly answering the visualization and prediction questions, as leaders are popular as the in-between points from the swarm to operator. Leaders can be used to gather aggregate information about the swarm to return to the interface, while also propagating user control inputs from the user to the rest of the swarm. Intertwined with the question of proper control and propagation of user commands is the phenomenon of *neglect benevolence*, which is the idea that swarms may need time to evolve before new commands should be issued. This will be discussed in Section 2.4.

Subsequently, I will discuss prior work on human perception of biological motion and common fate in Section 2.5, and how it may pertain to human-swarm interaction. While this research does not concern swarms directly, it is helpful for understanding how humans view collective motion and how such principles can be applied to HSI interfaces. I will then review the work in trust in automation—specifically trust in robotic systems—in Section 2.6. Finally, in Section 2.7, I will situate this work within the context of similar work—primarily that of network visualization and agent-based modeling.

2.1 OVERVIEW OF BIOLOGICALLY-INSPIRED SWARMS

Swarm robotics was originally studied in the context of biological swarms found in nature, but has since become its own distinctive engineering discipline [9, 4, 11, 127], since it promises to be useful in a wide range of potential applications including reconnaissance, environmental monitoring, tracking, exploration, search and pursuit-evasion, infrastructure support, convoy protection, and even space exploration [131]. Despite their potential, most robot swarms are still confined to laboratory settings and simulations; however, the biological metaphor has been useful in making improvements in the applicability of swarms for spatially distributed behaviors, as well as bringing swarms closer to the real world.

Biological systems have long since been an inspiration for the design of robotic systems in terms of hardware [52] as well as behavior [3]. Much of the work on swarm robotics originated from the study of biological swarms and swarm intelligence [9]. A recent survey [11] reviewed swarm engineering efforts and identified four areas that require further attention to support the development of real-world applications, namely (1) modeling and specification of requirements, (2) design of swarm behaviors, (3) verification and validation of swarm models, and (4) human-swarm interaction. The most interesting for the perspective of this dissertation is the fourth area, concerned with human operation and maintenance of swarms. In this area, particular concern is given to enabling effective control when lacking a centralized instance.

One of the better-known examples of a swarm algorithm derived from a biological inspiration is presented in [22]. Therein, Couzin et al. model the spatial behavior of animal groups with simple local interaction rules. These rules are determined by three parameters, the radii of three zones, namely zones of repulsion, orientation, and attraction. In the paper above, this simple model can generate four qualitatively distinct swarm behaviors: 1) swarm, 2) torus, 3) dynamic parallel, and 4) highly parallel. Which of the resulting behaviors a swarm generates depends on the choice of parameters and initial conditions, and raises the obvious question on how a human operator could interact with such a biological swarm model to induce transitions between these four types or change the direction of motion for a given type. This question was investigated in [43] through the injection of leaders and predators under the control of an operator, a paradigm that will be discussed further in Section 2.3.4.

Couzin's initial work has been extended in multiple ways. For instance, in [129] researchers show how interactions between agents in a collective can give rise to emergent group-level search behavior through the use of context-dependent interactions and dynamically selected leaders that are selected automatically to guide the entire swarm to the source. In [117], the researches alter Couzin's model to allow for adaptable control laws for each agent. In other words, the agents' motions are not only impacted by their current state and neighbors, but also the surrounding environment and current state of the task. For instance, if an agent perceives itself as moving in a good direction (e.g. toward the goal), the influence of neighbors on their heading will be decreased (so as to not steer them away from their good direction). The authors show that this greatly improves collective motion in complex environments, and achieves similar performance in non-complex environments (where the adaptive nature of the laws has less of an effect).

Another paradigm that falls under bio-inspired metaphor is pheromone-based communication [112, 113]. Pheromones have been used in [98] to coordinate a swarm to for surveillance, reconnaissance, hazard detection, and path finding. On a more general note, in [124] Sumpter identifies several principles that describe biological systems that exhibit collective behavior. Applying these principles to engineered systems has led to a wide range of bio-inspired systems, some of which are surveyed in [119]. Despite these previous efforts, there is no existing work investigating how humans can interact with a pheromone-based swarm system.

A number of recent projects have made significant progress developing swarm hardware as well. The “Swarmanoid” project [27] developed a swarm of heterogeneous mid-sized robots, including the popular SWARM-BOT platform s-bot [81, 28, 92]. Other projects and experiments used available platforms including the Kobot [16], and Kilobot [106, 107]. These examples, along with the decreasing cost of sensors and robotic hardware, and improvements in the distributed control algorithms, suggest that real-world applications for swarms are within reach. To achieve this, a number of challenges remain to be addressed—primarily, the study of human interaction with such swarms. For the most part, swarms are expected to operate autonomously. But the presence of a human operator can be beneficial and even necessary since the operator can (a) recognize and mitigate shortcomings of the autonomy, (b) have available “out of band” information not accessible to the autonomy and that can be utilized to increase performance, and (c) convey changes in intent as mission goals change.

A further benefit of the bio-inspired paradigm is that it creates an easy way to assess and develop collective behaviors commonly seen in nature, so long as the natural example is well understood. Simple behaviors such as flocking and path finding are common in nature, and have become equally common for swarms. The three bio-inspired behaviors that will be the focus of the user studies in this thesis are flocking, rendezvous, and dispersion. How these are designed and implemented will be discussed in Section 3.1, but the following will review prior work investigating each of these behaviors specifically.

2.1.1 Rendezvous

Aggregation, hereafter referred to as rendezvous, is one of the most simple and easy to recognize behaviors, is a process often found in natural swarm systems [15], and has been frequently adapted to artificial swarms [130]. A similar problem in control theory has been studied as the rendezvous problem [20]. The basic objective for both problems is to move all swarm robots towards a common location.

Bio-inspired aggregation behaviors have been implemented on real swarm robots in [130]. Therein the authors start with a model for a specific swarm robot, the s-bot, equipped with an omni-directional speaker, three directional microphones, and eight infrared proximity sensors. Weights for a neural network controller, with direct connections from every sensor to every actuator, are evolved under a fitness function that measures aggregation via the average distance of robots from the center of mass of the swarm. Two distinct aggregation behaviors were discussed: one leads to multiple static aggregates while the second leads to a single moving dynamic aggregate that resembles a flocking behavior.

Rendezvous does not necessarily entail the entire swarm aggregating at a single location. In some cases, an operator may want the swarm to aggregate in multiple disconnected locations, which is referred to as the multi-agent rendezvous problem. One research group in [63] and [56] demonstrates how the technique of glowworm swarm optimization can be used to guide a swarm to multiple disconnected rendezvous points. They provide both a theoretical analysis of their algorithm as well as a demonstration on real robots.

The rendezvous problem has been studied in [20] from the control theory perspective. Therein the authors define an abstract model of a robot that knows its own location and can transmit it to neighbors within its local communication network. The topology of the communication network is given by a proximity graph, and five different such graphs are discussed in [20]. The authors prove theoretical guarantees for the convergence of the swarm to the circumcenter under different static and changing communication topologies. The algorithm itself is fairly simple: each robot moves towards the circumcenter of the point set given by itself and all neighboring robots it communicates with, with the additional constraint that no existing communication links are to be broken by their motion. The analysis is more involved—particularly the work required to apply the LaSalle Invariance Principle [64] for a changing communication topology. The main assumptions for guarantees to hold are the ability to sense or receive the locations of neighboring robots and having an environment without obstacles.

Further work on the rendezvous problem has led to a reduction in the required sensor capabilities. For example, in [143] the authors present a solution to the rendezvous problem that does not require knowledge about the exact location of other robots, but instead uses only a quantized bearing sensor that reports the presence of another robot in a small range ahead of the robot.

2.1.2 Flocking

A more complex set of swarm behaviors is the formation of specific patterns of motions, specifically flocking, or consensus on a direction and speed of movement. One of the first algorithms to enable a swarm of robots to flock was presented by Reynolds in [104], with the motivation to simulate flocks of birds for computer graphics. Therein individual members of the swarm would follow simple local rules to avoid collisions (separation), match velocities to their neighbors (alignment) and center themselves amongst their neighbors (cohesion). Together these generate a flocking behavior. One of the earlier demonstrations of how to control a flock of animals, with robots influencing the flock, were presented in [133]. A simple controller for the robot was tested in a simulation with a swarm model similar to [104]. In [33], work on flocking is applied and implemented on robots with particular emphasis on the translation of control inputs to robot motion. More precisely, the force vectors resulting from the flocking rules for cohesion, separation, and alignment are translated into forward and angular velocity. The experiments in [33] show improved effective travel distance when considering magnitudes of the forces.

An overall framework for the analysis of flocking algorithms, including analysis of swarm fragmentation, is presented in [93], following a line of work from [128] and [54]. One of the most interesting aspects of [93] is the first introduction of a formal definition of what constitutes flocking. This definition is established with regard to 1) how much the flock differs from a lattice (i.e., a formation with all neighbors having a desired distance to each other) in terms of a deviation energy, 2) to what extent velocities are matched, 3) connectedness and cohesiveness of the flock.

Through the use of parameter manipulation, Couzin et al. [22] has shown that multiple types of flocking can be achieved with very small tweaks. The four types demonstrated in his work are a) swarm, b) torus, c) dynamic parallel, and d) highly parallel. The first, swarming, is the most chaotic, and can be envisioned with a swarm of bees hovering over a nest. A torus is simply a swarm flocking in a circle, and the two parallel types are what are traditionally thought of as flocking, with different degrees of rigidity between members (e.g. dynamic parallel is more like a loose flock of birds, whereas highly parallel can be envisioned with a tight school of fish).

2.1.3 Dispersion

Deployment of swarms, i.e., swarm dispersion governed by local control laws, is a swarm behavior typically used for area coverage. Swarms are expected to be ideal for area coverage, because this task requires covering, with sensors, a large area in order to observe some phenomena of interest or discover and track targets. One of the first to apply a force metaphor (a physics-inspired perspective similar to bio-inspired metaphors) for the distribution of large robot teams are Howard et al. in [53]. Therein, robots are repelled by obstacles and other robots and, as a consequence, distribute throughout an environment with obstacles. Experiments with 100 robots show successful dispersion in a realistic office environment and convergence to a static equilibrium.

A different approach to area coverage, with the goal of seeing every part of an environment, is taken in [37]. Therein the environment is given by a polygonal boundary and robots cover the environment by creating an incremental partition of the environment as they progress to cover it. Some results regarding convergence time and guarantees for a given number of robots are provided. A fleet of fifty-six real robots was used in [79] to test and compare five area coverage algorithms showing significant differences between the time to reach various goal locations and to fully disperse in the entire environment. This work also demonstrated how robots could self-charge, by keeping a pathway back through the communication network to guide each agent to a charging port.

2.2 VISUALIZATIONS OF SWARMS AND OTHER COLLECTIVES

The majority of research on HSI, including the research presented in this thesis, focuses on remote interactions (i.e., when the human operates separately from outside the swarm). For such interactions, the dominating constraint for improving visualization and prediction is that of communication, usually between a user at a computer terminal and a swarm operating at some distance from the human. This is because issues of limited bandwidth and high latency effect how information can and should be presented, and a human operator will need to account for these communication difficulties. Furthermore, issues such as operator workload and situational awareness can also be more difficult for a remote swarm, as the user is not placed within the context in which the swarm is operating. Further challenges regarding communicating information to an operator and the effect of resulting uncertainty from incomplete information are discussed in [48].

Despite difficulties inherent in remote HSI interaction, it is likely to be the default option for swarms that are entering otherwise inaccessible or dangerous areas. In fact, one of the key motivations for using swarms in real-world applications is their ability to be deployed in locations where humans cannot travel. Therefore, one of the primary challenges of HSI is to reconcile the distributed nature of swarms with a central human element of control and the ability to collect information about the swarm and environment. Part of this is a technical challenge, addressed in the study of sensor networks [142, 1] and mobile ad hoc networks [5, 100]. It is noteworthy that many swarm algorithms are also used in managing sensor networks to overcome bandwidth and latency issues [101] and design routing protocols [109, 57].

Nonetheless, properties of the communication channel impact how the swarm can and should be visualized. Proper visualization must balance the limitations of the communication channel between the human and swarm with the needs of the human operator. Here, particularly, the question is how one can use the bandwidth available to display as much information as possible to the operator, in a manner that is easily understandable. Some previous work in HSI has addressed this from an operator control perspective [138, 90].

2.2.1 Communication Restrictions

A significant difficulty faced by designers of a human control interface for swarms is not only to visualize the swarm state properly, but also to facilitate the understanding of swarm dynamics as well as the impact of control inputs. As mentioned previously, the swarm model being employed (e.g. bio-inspired), can offer a helpful metaphor for this problem, and should be incorporated into the visualization. For example, a visualization of forces might aid comprehension of a biologically- or physics-inspired swarm, especially for an operator familiar with attractive and repulsive forces. Very little research, however, has investigated these ideas. We already know that visualization of a swarm is particularly difficult for the operator in situations with incomplete information. Such situations arise in the real world from constraints on bandwidth and communication latency that arise in operations taking place in remote locations, as well as from sensing errors and uncertainty.

To illustrate this concept using a simplified example, imagine a remote environment where high bandwidth communication back to the human operator is not possible and where communication latency between the swarm and operator would be significant, but where communication between swarm members has relatively low latency due to their close proximity. Now imagine that an operator must control a 200-member swarm to perform some task in this environment. If we assume a two-dimensional plane, at each update step every robot would need to report, at minimum, four variables back to the operator: x-coordinate, y-coordinate, heading, and velocity. If each of these variables is represented as a 16-bit variable, a single update step would require 12,800 bits to be transmitted. While this is well within current capabilities for some remote locations, such as the Moon, it still represents a significant bandwidth requirement if updates need to be performed multiple times per second, and would be well beyond capabilities for many currently unexplored environments, such as deep ocean exploration. Simply establishing an independent connection between each swarm member and the human operator working at a base station would be difficult, as messages could be received out of order and there could exist significant crowding of frequencies as the number of swarm members that can be fielded grows. Moving on to latency, our current understanding of physics places a floor on how low latency can go for a given location, bounded by the speed of light. If, for example, we wanted to operate a swarm on Mars, the lowest latency we could achieve is between 3 and 22 minutes depending on the relative position of the Earth and Mars in their orbits. This makes immediate feedback and human-swarm control as we normally imagine it impossible. Therefore, we must design interfaces and visualizations that can help operators cope with these communication restrictions.

As of the writing of this thesis, there are very few works that investigate overcoming bandwidth and latency issues in swarms; however, those that do exist can provide a baseline for future research. The first study investigating this problem viewed how a predictive display can aid operators in controlling a flocking swarm searching for hidden targets in an open environment [138]. In this study, participants were presented with one of two conditions: a naïve display, where each robot's position and heading were shown to the operator, and a predictive display that, in addition to the naïve display, showed a prediction of where each robot would be if they continued on their path for $2x$ seconds, where x was the latency between the swarm and operator (10 seconds in this study). This means that the information shown on the display without prediction was 20 seconds old (10 seconds of latency in each direction). While far from perfectly accurate, this simple predictive visualization allowed participants to find significantly more targets than those without the prediction. Both cases, however, fared worse than a control condition in which no information latency was present. The results of this study were the first to show that properties of the HSI system—in this case, properties of the communication channel between the swarm and operator—could be incorporated into the display to aid operator performance. This study was also the first to highlight the phenomenon of neglect benevolence—a concept that will be discussed in Section 2.4. Missing from this study, however, was a critical analysis of different manners in which to display this prediction. For instance, taking into account robot interactions when making the 20 second prediction may give improved performance. Similarly, displaying summary information about the predicted state of the swarm (e.g., an estimated centroid and ellipse containing the predicted positions of the swarm members) may be more intuitive for an operator. This thesis will attempt to address these shortcomings by testing different visualizations in an interface for similar tasks.

A similar study was performed in [90]. Here, the researchers looked instead at bandwidth restrictions in the swarm to human communication channel. Presented to participants were one of three different bandwidth conditions (and consequently, three different displays) in random order. The control condition, also called the full information condition, gave the operators real time updates of each member of the swarm as it moved through the environment. A second condition, called the medium bandwidth condition, assumed that, while swarm to human communication bandwidth was restricted, bandwidth between swarm members was not. Therefore, the swarm could collectively aggregate information and return summary information to the user's display. This took the form of a centroid, showing the average position of the swarm in the environment, as well as a bounding ellipse, which showed the average variance in the x- and y-axis of the swarm. Finally, the low bandwidth condition assumed that both the human-to-swarm, and the inter-swarm, communication channels were restricted. Here, the user would see only one robot's position updated each second. The final results showed that the summary display (medium bandwidth condition) was just as helpful as a display with full information, in terms of total targets found. The low bandwidth, single-update display performed significantly worse. The takeaway here is that full information may not always be required for the best performance. Indeed, one can think of situations in which it may actually be harmful to performance. For instance, when dealing with larger numbers of robots (only 40 were used in this study), the operator may quickly become overwhelmed with full information, and unable to grasp the global behavior occurring. This thesis will further investigate these findings.

2.2.2 Other Visualization Research

Recent work [12] showed that small samples of angular velocities and concentration of neighbors can be sufficient to automatically classify the behavior of a swarm following a common flocking algorithm [58] as either flocking (moving in a common direction) or torus (moving in a circle). Reducing the amount of noise and aggregating and fusing information to simplify the problem of determining a swarm's state are promising research areas in swarm state visualization. Other researchers have used multimodal feedback to improve operator control of swarms. In [46], the authors used a potential field approach for controlling the swarm for a convoy protection scenario, and designed an interface that provides feedback regarding the swarm speed, strength, capability, and dispersion. The feedback was presented as visual, auditory, tactile, or a combination thereof. The results showed that both operator performance increased and workload decreased with participants who used the multimodal displays instead of a single display (e.g. visual only).

Besides the aspect of designing appropriate algorithms that provide aids to humans for swarm state estimation, there is the very important issue of whether humans may be able to learn to understand swarm dynamics, given appropriate feedback. This question has hardly been investigated, and is essential for operators that wish to change or properly assess swarm behavior. In [126], the authors investigate whether human operators can learn to predict the effects of different input behaviors to a simulated swarm. The authors use a two-choice control task, whereby operators choose either a dispersion or a rendezvous algorithm for a swarm randomly distributed in an environment with obstacles. The goal was to cover as much of the environment as possible in each trial. Results from the experiments showed that human performance increased over the 50 trials from an average of 60% to 80% correct, thus indicating that humans could learn to estimate the results of deploying a particular behavior on task performance. The results of this study are interesting from another perspective as well, because they were used to create a computational cognitive model of the human operator that mimicked the human performance [125]. This is so far the only study using a cognitive architecture to model human operators in an HSI task.

Finally, in [85], the authors investigate whether human operators can acquire enough understanding of swarm dynamics to predict the effects of the *timing* of their control input. In this study, operators were tasked with observing a swarm moving from a random initial state to some first formation, and determining the optimal time to give an input signaling the swarm to move to a second, goal formation. One condition showed a naive display, where only the positions of the robots were displayed as they moved. A second condition gave the operators information about the robots' velocities and headings to aid in prediction. In both cases, operators had to give the input at the point that would minimize the convergence time to the second formation. However, due to the phenomenon of neglect benevolence (see

Section 2.4), the optimal input time was not necessarily as early as possible. The argument in [85] is that an aided display is important in such cases because it is difficult to perceive the optimal input time by simply looking at the emergent behavior of the swarm—a hypothesis shared by this thesis. The aided display, informed by the control algorithm, seemed to help operators overcome this issue.

These studies show promise, but existing approaches to visualizing a swarm's state are largely situational, and set up based on what the designers think will improve the quality of human control. Instead, research should focus on evaluating different visualizations based on the current swarm behavior and properties of the communication channel, and taking a more human-centered approach by using existing literature in perception of biological motion and common fate (see Section 2.5). Evaluations can use specific performance metrics—from quality of operator predictions to time to convergence toward an operator's input. This thesis will attempt to address this shortcoming by comparing different visualizations to determine which ones improve human control, and also what different swarm behaviors and tasks require in terms of visualization.

2.3 SWARM CONTROL METHODOLOGIES

We will now focus on the other side of the control loop: how to properly convey input from the operator to the swarm. Due to the fact the human control of swarms is desired to be equally effective regardless of the swarm's size, it stands to reason that in many cases a swarm can be viewed as a single entity, much as a system with one robot and one human would be, except that the properties and behavior of this system would be different than that of a single robot. This may not always hold, as some swarms contain heterogeneous members, and some will require splitting into disconnected parts or giving different members of a swarm different commands. Therefore, there is a need to better formalize the types of control an operator can exert on the swarm. In this section, I will discuss the following types of swarm control, which make up the vast majority of current research in HSI:

1. Switching between algorithms that implement desired swarm behaviors,

2. Changing parameters of a single swarm control algorithm,
3. Indirect control of the swarm via environmental influences, and
4. Control through selected swarm members, typically called *leaders*.

Within these swarm-specific types of control, a distinction will be made between continuous and discrete inputs. For example, leader-based influence can be achieved with a continuous input to a leader (teleoperation) or with a discrete input (leaders move toward point [x,y]). The above types are also not mutually exclusive. For example, leader-based control can be used in conjunction with any of the other three control methods. Furthermore, each of these control methods can interact with other properties of the human-swarm system, such as the communication scheme, and impose varying constraints on the swarm.

2.3.1 Algorithm and Behavior Selection

Control via algorithm and behavior selection assumes that the human operator is giving control inputs at discrete time points by selecting a specific swarm algorithm, such as those discussed in Section 2.1. It also assumes that operators have at their disposal a library of algorithms that implement different swarm behaviors. By choosing different algorithms, human control is akin to controlling hybrid systems with the human acting as a switch. During the time that a behavior is active, an algorithm, usually a local control law, implements the behavior autonomously. A comparison between behavior selection and environmental influence in [61] indicated superior performance for behavior selection for novice operators, although the same may not be true for operators more familiar with the swarm and how it behaves. Behavior selection was also used in [137] and [69] for swarm control. Successful control with behavior selection also presupposes that the operator can develop an understanding and has access to an appropriate visualization of the swarm dynamics [85]. In a number of studies, operators used this control type to select the behavior of a selected group of leaders. This mixes the control via leaders, discussed in Section 2.3.4, with behavior selection.

Overall, control via algorithm or behavior selection appears to be an effective method of swarm control when the robots have a high degree of autonomy and can operate largely without error or human oversight between human inputs. Once instructed to execute a certain behavior, an operator largely relies on the autonomy of the swarm as well as the autonomy of individual robots to deal with obstacle avoidance, robot-to-robot communication, and local coordination. The transmission of commands from the operator for this type of control also does not usually pose significant constraints on the communication network, as they are infrequent. The greater challenges here relate to the selection of the right behavior, input timing, and state estimation—the operator needs to understand what different swarm behaviors look like in order to employ proper selection and switching.

2.3.2 Control via Parameter Setting

Most systems depend on a set of parameters for their operations, and so can many swarm algorithms. The values for these parameters offer a clear avenue for control and influence for an operator, in both discrete and continuous input settings. The key difference for swarms, as opposed to single- or multi-robot systems, is that parameters do not directly influence the behavior, but rather have indirect effects through behaviors emerging from interactions within the swarm and its environment.

In [22] the wide range of behaviors that can be generated with a simple flocking algorithm given different parameters is presented in great detail. These insights have not yet lead to a human-controlled transition between emergent behaviors by changing the parameters of the system, however. Another study that considered the setting of parameters is found in [59], yet it focused on indirect parameter setting aided by an autonomous algorithm rather than allowing an operator to directly modify parameters. Therein, Kira and Potter present preliminary work for a top-down and bottom-up approach for a physics-based swarm control system. For the top-down approach, an operator can set desired global characteristics, such as swarm radius and maximum inter-agent distance (i.e., a parameter setting interaction). For the bottom-up approach, virtual agents (point particles) are placed in the swarm and interact with it via simulated gravitational forces. Evolutionary computation is then used to learn an appropriate placement and parametrization of these virtual agents to bring about a particular behavior (e.g., a split into two groups). An operator does not directly interact with the swarm parameters or inject virtual particles in these cases, but instead relies on the learning mechanism as an intermediary. Here, placement of the virtual particles resembles an environmental interaction (see Section 2.3.3). The algorithms were tested on a “defend a resource” scenario first in simulation with one resource, six agents, and three virtual particles, and also on six Pioneer robotic platforms in the laboratory. No experiments with human subjects have been reported regarding the effectiveness of this approach.

Another example of parameter setting to control a swarm is found in [51]. Therein, an operator controls a swarm of UAVs, in simulation, by setting the parameters for the “personality” of UAVs, defined by four characteristics: conformity, sociability, dedication, and disposition. These relate to thresholds in a target assignment and bidding process. In addition, the operator can designate regions in the environment as either hot or cold. Hot regions are suggestions to nearby UAVs that this region will contain targets while cold regions suggest the opposite. Whether a UAV incorporates the operator’s suggestion depends on its conformity. There were no user studies carried out in [51], nor any results presented. Some results for a similar system are found [50], but are also lacking user studies.

Despite the examples shown above, parameter setting is most often done during the design state of the swarm, and particular parameters that enable an operator to generate multiple emergent behaviors are often desired. An example of this is found in [13]. Therein, the authors investigate the parameter space for a flocking algorithm to determine a set of parameters that allows flocking and torus formations to emerge. An operator then influences a subset of the swarm via teleoperation to switch between flocks and torus formations. The results indicate that it is easier for an operator to switch from a torus to a flock when the teleoperated robots influence the rest of the swarm via their orientation. These results were obtained using simulation runs in a “Oz of Wizard” style study [121], i.e., with simulated human input.

Many contributions with user studies use some combination of parameter setting with another control method, such as leader selection or behavior selection. For example, in [137], the authors allow operators to select between two different behaviors—disperse and “go-to”, the latter of which was similar to rendezvous, except with the rendezvous point parameter specified by the user. Similarly, in [136], the authors task human operators with moving a swarm guided by a leader to various goal positions around an environment by specifying the goal direction of the leader. This work was extended in [135] by using multiple leaders that varied over the duration of the experiment.

2.3.3 Environmental Influence

One of the distinctly "swarmish" interaction types is to influence a swarm through environmental factors. Environmental influence involves altering part of the environment, usually virtually, but sometimes physically, to influence the behavior of a swarm within that part. Environmental influence has been implemented as a variety of constructs, including virtual pheromones, virtual beacons, and amorphous computing. The key characteristics of this interaction type is that it is location-dependent and persistent through time (or slowly vanishing in the case of pheromones). Behavior selection in contrast sends a single instruction that can be independent of location and affects robots when it is received and subsequently propagated.

Environmental influence on the swarm is mediated via direct or virtual sensing of environmental changes. Robots in the swarm continue to operate under the same rules they were deployed with and interact with the environment in a consistent manner throughout their operation. It may be argued this is a more suitable way to control the swarm, as it does not directly interfere with the autonomous emergence of different swarm behaviors, i.e., if proofs can be given to show guaranteed emergence of a behavior under some distributed control algorithm, environmental control should not necessarily affect that guarantee. This, however, depends on the type of environmental influence available, particularly when using virtual pheromones and beacons, and whether the emergent properties are guaranteed in the particular environment.

An example of environmental influence is found in [24]. Therein, the authors use the analogy of a digital display to represent a swarm of robots, whereby each robot represents a "pixel" in the environment, and gives information only from its local environment and neighboring robots to a human operator. The example they give is that of a search and rescue scenario inside a building, where a deployed swarm can spread out and, once a victim is identified, the robot viewing them can propagate its information back through the swarm via virtual pheromones to the human operator. In their case, the rescuers can then view the combined information from all nearby robots on a head-mounted display as they travel through the environment looking for the victims. Furthermore, the human operator can influence the swarm by injecting pheromone information to nearby robots via a handheld display. Note that this is also a rare example of proximal interaction with a swarm, as opposed to the usual remote interaction. Another example of virtual pheromones is given in [139], wherein operators demonstrate the ability to use virtual pheromones to control up to 50,000 robots in simulation. Another example of environmental influence is given in [60] and [61], where the authors use simulated beacons that can be placed by an operator and signal to nearby robots to execute a certain behavior. A set of seven different behaviors are implemented. The beacons can be placed anywhere in the environment to allow the operator to modify the overall swarm behavior via the perceived environment as he or she sees fit. Experimental results indicate, however, that behavior selection for the same set of behaviors leads to superior performance, as compared to placing beacons, for untrained operators on a foraging task.

2.3.4 Leader Selection and Teleoperation

One method to deal with the complexity of controlling a swarm is to allow an operator to select and control a subset of the swarm, thereby reducing the number of robots that have to be considered simultaneously. Individuals or groups of robots selected by an operator are frequently denoted as leaders, since they are expected to influence and lead the remaining swarm as a proxy for the operator. The selection of a small set of individual robots as leaders opens up the possibility for more engaging forms of control that are also used for single and multi-robot systems, such as teleoperation. The key difference between swarms controlled via leaders and other systems is that leaders have an influence that propagates through the swarm, and an operator should attempt to control the entire swarm via this propagated influence. In multi-robot systems, the goal is rather to avoid such interaction effects to reduce the cognitive burden. In swarm systems one needs to exploit these interactions to the operator's benefit. The main questions for leader-based control are (a) how to best select the leader(s), (b) whether a selected leader remains a leader throughout a scenario or whether leadership is transient, (c) how to control for propagation effects on the remaining swarm, and (d) how leaders should interact with nearby swarm members.

2.3.4.1 Continuous Influence via Leaders In cases where more precise control over a swarm's operation is needed, or when a desired emergent behavior cannot be generated autonomously and without significant human influence, continuous inputs may be given by a human operator. These continuous inputs will have a persistent influence on selected leaders and indirectly on the swarm, and such situations require significantly more training and attention on the part of the operator. In its basic form, persistent influence is akin to teleoperation. It generally involves some notion of the state of the system fed back to the operator who can then modify the inputs accordingly. Such control usually requires a tight feedback loop with low latency and a representation of the system state that is interpretable for the operator. But proximal interactions are also conducive to continuous control since the human can be sensed by the robots continuously and can direct them much like a leader robot, and thus any movement of the operator is potentially an input to the swarm. Section 2.2 briefly discussed the difficulties of estimating and visualizing the state of a swarm. For controlling motion of single and multi-robot systems, visual and haptic feedback has been used predominantly, and these do not easily translate to swarms without modification. The selection of swarm leaders, however, can enable such control. In this case, the control of a single leader or a group of leaders is similar to single robot or multi-robot teleoperation. The key difference is the influence of the motion of swarm leader on the remaining swarm that has to be taken into account.

In [6] a leader robot in the swarm is teleoperated in order to aid in the localization of a radiation source. The swarm is influenced indirectly through the motion of the teleoperated robot. The influence is determined by the mode of the robot and can "push" other robots or direct them into one of four directions (up, down, left, right). Once deselected, the robot can be instructed to maintain its mode and thereby its influence on neighboring robots. Results of a small user study indicated that human-operated swarms were significantly better than a fully autonomous swarm at finding the radiation sources within the environment. Goodrich et al. [40, 41, 42, 43] have also worked extensively on leader-based control of swarms that follow Couzin's control laws [22]. Therein, the authors investigate using teleoperated leaders, which will either attract or repel neighboring robots, to allow a human operator to control the swarm. The authors also consider swarm members, so called stakeholders, that are influenced by the operator as well as other swarm members, in contrast to the teleoperated leaders (also called predators in the case of repelling leaders). An emphasis is placed on determining under what conditions operator influence can lead to different emergent behaviors and formations.

In [99], the authors implement a leader-based model both in simulation and on real robots, using both virtual agents and a human operator as leaders in a swarm, and found that this method scales reasonably well to larger swarm sizes in an information foraging task without obstacles. In [136] the authors propose two methods for propagating operator intent from a single leader to the rest of the swarm. The first is *explicit*, where the leader can be distinguished from other neighboring robots, and thus its neighbors can explicitly follow the leader's direction; and the second is *tacit*, where the leaders are indistinguishable, and implicitly bias the average speeds and headings of neighboring robots. Here, the authors found that the explicit method gave human operators better control over the swarm, but hypothesized that the tacit method could be more robust to sensing error if a larger percentage of the swarm were leader robots to allow for faster propagation of used intent. In [135, 134], the authors further this work by presenting an algorithm for selecting multiple leaders dynamically in a swarm as the topology of the communication graph changes. It was found that, while the explicit method of propagation was again superior overall, the tacit method did indeed perform better under significant sensing error.

2.3.4.2 Discrete Influence via Leaders Numerous works have implemented discrete control systems in which the operator sends messages to selected robots intermittently. This method is easy to implement and requires little training for the human operator. It is also well suited for both homogeneous and heterogeneous swarms, as different commands can be easily and distinctly given to each type of robot. For example, in [69], operators effectively deployed a heterogeneous swarm in an ocean setting to test the viability of swarms in monitoring data in waterways. The operators had sparse, intermittent communication with the robots—

being able to send and receive data only when the robots surfaced, and sending commands to correct errors in the robots' trajectories due to sensing error and ocean currents.

In [21] the authors present a method for a user operating amongst the swarm to select and assign tasks to a single leader robot out of many in an indoor environment (with a distance between the human and robot between 1 to 4 meters). Each robot first recognizes how directly the human is looking at it through facial recognition. It then uses a ring-based leader election algorithm to determine the single robot with the highest face detection score. The user then commands this robot with gestures. Pilot experiments with human participants produced encouraging results, yet it is not clear how the approach scales and how appropriate it is for larger distances between user and robots. Also suitable for discrete control inputs is the work presented in [39, 88, 103], which enables proximal interactions with operators by transmitting commands to the swarm with gestures, face engagement, and speech.

2.3.4.3 Teleoperation The selection of single leaders or small groups of leaders has been the default choice for much of the work on HSI that involves persistent and continuous influence. One of the few exception is found in [91]. Therein operators used a haptic joystick to give continuous inputs to the entire swarm during a target searching-task. The human teleoperated the swarm via broadcast commands by manipulating the joystick. The swarm itself handled obstacle avoidance and maintenance of proper robot-to-robot distances, but global goal direction and speed of the robots was controlled by the human. The haptic feedback given to the operator is computed as the average of all forces exerted on all swarm robots resulting from repulsion from obstacles, similar to the approach in [53]. The authors found that giving continuous inputs with haptic feedback allowed for superior control and more targets found.

In general, teleoperation of robots has been studied extensively, but the primary emphasis has been on single robots. I will now review some of the work done for bilateral teleoperation of multi-robot systems, for which there is usually a *master* robot that a human uses to control a *slave* robotic system. Information is fed back and forth (as forces) between the human and the slave system through the master robot or haptic device. Haptic feedback can be used to augment existing methods like continuous visual feedback. Recent efforts in this area are found in [118, 105, 35]. These can be broadly put in two categories depending on the communication (and control) architecture between the master and slave systems: (a) Centralized approaches - where each robot communicates individually with the master system [118, 105, 17] and (b) Decentralized approaches - where the robots coordinate among themselves and only a single robot communicates with the master robot [35, 114, 66, 36]. Control and communication should ensure safety and stability, i.e., avoid collision and track a desired reference trajectory (e.g., maintaining a certain formation).

A decentralized strategy was proposed by Franchi et al. [35] based on a leader-follower approach where the slaves are assumed to have second order point mass dynamics. The key contribution is to design a potential function (and hence a control law) that ensures that the overall system passive. The controller has been tested with a human controlling a team of up to six simulated UAVs. Although the authors allow the agents to make or break links, there is no guarantee that the connectivity of the robotic network is maintained. In [38] and [114], the authors have extended the work from [35, 66] to ensure that the designed haptic control laws ensure stability even in the presence of delays. Similar techniques have also been used for haptic control of UAV formation where the UAVs only use the bearing information of their neighboring agents [36]. The above control schemes have been limited to either formation control or target tracking. Haptic control schemes for other multi-robot tasks (e.g., area search and coverage, foraging, cooperative mapping, etc.) are not available. To apply this work on teleoperation to a swarm, in particular a selected subgroup of leaders in the swarm, the repercussions and effects of the motion of this subgroup on the overall swarm behavior and dynamics would need to be integrated into the control scheme so that an operator can control the subgroup while being aware of the implications and compensating for the overall swarm behavior.

2.4 NEGLECT BENEVOLENCE

Not only is the method of giving different commands of concern to human operators and those designing the HSI system, but also the timing of those commands. Since some swarm algorithms require time to converge and stabilize after an operator command is issued, it is possible for the same types of commands to have different—sometimes adverse—effects depending on the state of the swarm. To capture the idea that humans may need to observe the evolution of the swarm state and wait some time before acting, a novel concept called *neglect benevolence* was investigated. This concept is in some sense the converse to *neglect tolerance* [94, 120] in human-robot interaction (HRI) of independent and non-coordinating robots, where it is assumed that the performance of an individual robot degrades with time, and hence the attention of the operator needs to be scheduled so that the time between servicing robots (the neglect time) is minimized [75].

To illustrate the phenomenon of neglect benevolence, consider a simple flocking algorithm. One of the issues that may occur for flocking is the fragmentation of the swarm, and frequent instructions for changes in direction of the flock may lead to such fragmentation unless the swarm regains its cohesion before the next instruction. The risk of fragmentation is increased by delays in coordination, errors in motion, and sensing as well as external perturbations. In [87], it was shown that improper timing of control input could lead to swarm fragmentation. Also, in [138], the authors show evidence of neglect benevolence in swarms during a simple target-searching task. In that work, operators who issued commands frequently showed lower levels of performance than those who allowed the swarm to adjust between new commands. This was the first study to give evidence to the concept of neglect benevolence by showing that commands given too frequently to a swarm exhibiting emergent behavior could actually degrade performance.

Neglect benevolence is formally defined in [87], where the authors proved the existence of neglect benevolence for linear time invariant systems, developed an algorithm to determine the optimal input time for such a system. In [85] the authors further investigate human performance in the face of neglect benevolence and showed that human study participants learned to approximate the optimal time over the course of the experiment in a formation control task. Neglect benevolence and optimal timing studies are just beginning to emerge and they are an interesting area for future research. Additionally, algorithms to determine optimal human timing could be incorporated to provide operator decision support.

Proper visualization of a swarm is key to understanding how to time proper inputs for an operator. Neglect benevolence, being a somewhat opaque phenomenon, requires smart visualization in order for naïve users to understand, as it is difficult to time commands properly if the current state is not understood fully. The work in this thesis will aid in future research on neglect benevolence by investigating how researchers can construct correct visualizations which allow for the best understanding and predictive abilities of swarm operators.

2.5 BIOLOGICAL MOTION

In order to address human recognition of biologically-inspired swarms, I will also take inspiration from previous work in perception of biological motion. Human perception of biological motion has been investigated extensively since the 1970s. While humans are generally good at picking out motion, we seem especially adapted to picking out biological motion. In [55] the authors present the seminal work on perception of human such motion, showing that participants exhibited perfect accuracy when discerning a walking human given only 10-12 points of light (i.e. joints, etc.). The authors argue that this is an inherent ability in humans, and not one that is learned from the environment. Furthermore, they argue that humans can not only perceive humans walking given such limited information, but also subtle differences in gait, such as walking vs. running, limping, etc.

The argument that such ability is an intrinsic property of the human visual system is further supported by later research. In [34], the authors show that infants as young as 4 months old give preference to this sort of biological motion over other types of motion. Other research has shown the ability of participants to discern differences between male and female gaits [74], and between different emotions [26]. Since the early days of this research, the idea that humans are especially good at identifying human-like motion, and to a lesser degree animal motion [102], is well-established, and areas of the brain where this sort of recognition occurs have been identified [45, 95].

This work does not necessarily mean that humans will be good at recognizing all kinds of biological motion, however. Of particular importance to this thesis is recognition of *non-human* biological motion. To this end, some researchers have investigated human perception of common fate, or how well humans can discern groups of agents moving in a linear manner against background noise [132, 123]. However, this work is limited to linear motion and its limits (i.e., how far from parallel can dots stray before they are not identified as a common group). The research has yet to determine how well humans can combine this with the work mentioned previously. In other words, how humans perceive collective biological motion—the type of motion that would be present in swarming robots. Such research is an essential part of HSI, as it determines how well human operators are able to determine what behavior the swarm is performing, and thus how much information is sufficient for the swarm to return to the human to allow for proper recognition. This thesis will begin to address these points.

2.6 HUMAN TRUST IN AUTOMATION AND SWARMS

In human interaction with automation, it has been observed that the human may fail to use the system when it would be advantageous to do so. This has been called *disuse, underutilization,* or *under-reliance* of the automation [96]. People also have been observed to fail to monitor automation properly (e.g. turning off alarms) when automation is in use, or they accept the automation's recommendations and actions when inappropriate [72, 96]. This has been called *misuse, complacency,* or *over-reliance*. Disuse can decrease automation benefits and lead to accidents if, for instance, safety systems and alarms are not consulted when needed. Another maladaptive attitude is automation bias [83, 30, 65, 77, 111], a user tendency to ascribe greater power and authority to automated decision aids than to other sources of advice (e.g. humans). When the decision aids recommendations are incorrect, automation bias may have dire consequences [82, 84, 2, 78] (e.g. errors of omission, where the user does not respond to a critical situation, or errors of commission, where the user does not analyze all available information but follows the advice of the automation).

Researching and improving visualizations of swarms without also taking into account user trust on the swarm system would be misguided. In the literature on trust in automation, there is the concept of *transparency*, which looks at how the information provided to the operator by the autonomous system affects their trust in said system. According to the commonly-cited models in [67] and [76], the extent to which a human can understand the way in which an autonomous system works and predict its behavior will influence trust in the system. There is far less research on effects of transparency, with most involving level of automation manipulations. An early study [68] in which all conditions received full information found best performance for an intermediate level of automation that facilitated checks of accuracy (was transparent). Participants, however, made substantially greater use of a higher level of automation that provided an opaque recommendation. In that study, ratings of trust were affected by reliability but not transparency.

More recent studies have equated transparency with additional information providing insight into robot behavior. Researchers in [10] compared conditions in which participants observed a simulated robot represented on a map by a status icon (level of transparency 1), overlaid with environmental information such as terrain (level 2), or with additional uncertainty and projection information (level 3). What might appear as erratic behavior in level 1, for example, might be "explained" by the terrain being navigated in level 2. Participants ratings of trust were higher for levels 2 and 3. A second study manipulated transparency by comparing minimal (such as static image), contextual (such as video clip), and constant (such as video) information for a simulated robot team mate with which participants had intermittent interactions but found no significant differences in trust. In [140], researchers took a different approach to transparency by having a simulated robot provide "explanations" of its actions. The robot guided by a POMDP model could make different aspects of its decision making such as beliefs (probability of dangerous chemicals in building) or capabilities (ATR has 70% reliability) available to its human partner. Robot reliability affected both performance and trust, and explanations did not improve performance but did increase trust among those in the high reliability condition. As these studies suggest, reliability appears to have a large effect on trust, reliance/compliance, and performance, while transparency about function primarily influences trust only.

Because the concept of trust in autonomy will increase in importance as robots and swarms become more prevalent and more complex, the final study presented in this thesis will also look at how improving or changing the visualization of a swarm (transparency) can influence operator trust in a semi-autonomous swarm.

2.7 PLACE WITHIN LARGER CONTEXT

It is my hope that this research is applicable not only to researchers in HSI, but also those working in related fields. I believe there is much to be gained from situating this work within the context of similar fields. Specifically, there is significant connection between this work and research investigating how to visualize wireless sensor networks [110], as well as network graphs for social networks [49, 29] and research communities [25]. The primary difference between this work and those mentioned, however, is that HSI involves visualizing a moving network of robots, rather than a static display of a social network, for instance. Similarly, HSI typically requires time-sensitive human input, which requires constant human attention and vigilance, whereas static networks typically do not.

Despite these differences, there are multiple insights from these works that can inform the work of this thesis. For instance, in [29], the authors demonstrate that human recognition of overlapping groups, or clusters, within networks can be improved with smarter visualization techniques, such as highlighting cliques with different colors and convex hulls, and that these visualizations should be made interactive to allow for changes by users on the fly. Similarly to the work proposed in this thesis, [110] show that wireless sensor networks with severely limited communication abilities can be visualized intelligently using protocols built in to the ZigBee framework and a dynamic smartphone application.

The growing research on agent-based modeling (ABM) also has a large intersection with this work in HSI. Agent-based modeling uses independent virtual "agents" to simulate a dynamical system. The drawbacks of this approach over more traditional mathematical models are similar to the drawbacks in robotic swarms, in that global behavior cannot be specified, and proofs guaranteeing convergence or certain outcomes are difficult to provide.

However, the benefits of ABM are that they are extremely useful for simulating systems with emerging properties and many independent interacting agents, such as human systems [8], economics [32], and biology and ecology [115, 44]. This fact gives the field of ABM significant overlap with robotic swarms.

Of particular interest for this thesis is the work in [62], which focuses on improving the design of visualizations in ABM. Therein, the authors point out that much research in ABM neglects the visualization side of their model, making presenting their results more difficult than it needs to be. Particularly, they point out that commonly-known work in perception and visualization, such as Gestalt principles and graphics, should be used in conjunction with ABMs to improve the reach of their work. This is true for visualization of swarms in HSI systems as well, and thus is an important influence on the work in this thesis.

3.0 SYSTEM DESIGN

The following chapter will introduce the common framework built to conduct the user studies presented in this thesis. This first includes the control algorithms used to generate the three behaviors employed in the user studies in Section 3.1, followed by an overview of the CUDA-based swarm simulator in Section 3.2 and the two methods for selecting swarm leaders in Section 3.3: random competition clustering (RCC) and minimum volume enclosing ellipse (MVEE). Following, I will conclude this section by introducing the four visualizations used to display the swarm in Section 3.4. While each of the user studies is based on this framework, there are some small differences between them. Such differences will be explained within their corresponding user study sections in Chapter 4 where appropriate.

3.1 CONTROL ALGORITHMS

Before discussing the general architecture of the simulator, it is necessary to introduce the underlying control laws for each of the three behaviors that will be employed in the presented user studies. As discussed in the introductory chapter, the behaviors are: rendezvous (Section 2.1.1), flocking (Section 2.1.2) and dispersion (Section 2.1.3). Figure 3.1 gives examples of each of the behaviors, along with the neighborhood regions used. The presented algorithms use several common parameters that are defined in Table 3.1.

Table 3.1: Common simulation parameters for all user studies.

Variable	**Description**
d_1	Close range (meters)
d_2	Mid range (meters)
r	Maximum range (meters)
v_{max}	Maximum velocity (meters/second)
α_{max}	Maximum angular velocity (radians/second)
w_a	Align vector weight
w_c	Cohere vector weight
w_r	Repel vector weight

3.1.1 Rendezvous

The rendezvous behavior is executed by computing two vectors each time step: the repel vector, $\vec{r_r}$ and the cohere vector, $\vec{c_r}$. Vector $\vec{r_r}$ is defined as the sum of vectors from each neighbor robot within d_1 to this robot's (x, y) position. Vector $\vec{c_r}$, is defined as the vector from the robot's position to the center point of the rectangle $R = (x_{max}, y_{max}, x_{max}, y_{max})$, where x_{min} and y_{min} are the minimum x and y coordinates of the robots in the neighbor set N within r, respectively, and x_{max} and y_{max} are the maximum. The computation of this cohesion vector is adapted from the parallel circumcenter algorithm for rendezvous in [14]. The final goal vector is then computed using the following equation:

$$\vec{g} = w_r\vec{r_r} + w_c\vec{r_c} \tag{3.1}$$

3.1.2 Dispersion

The dispersion behavior is computed using only one component vector, the repel vector, $\vec{r_d}$, which is computed by taking the sum of the vectors from each robot in N within the maximum range r. Note that this is the same as the repel component of rendezvous, except that the robot will repel from any neighbor within maximum range r, not the closer range d_1. The goal vector for dispersion is equal to $\vec{r}_d$. Note that the result of this control law set is a swarm where each robot is exactly at distance r from each of its neighbors.

3.1.3 Flocking

For the flocking algorithm, control leaders are selected according to the distributed RCC leader selection algorithm, presented in Section 3.3.1. The non-leader robots each compute a repulsion vector $\vec{r}_f$ identically to the rendezvous algorithm (Section 3.1.1), except using d_2 as the maximum range instead of d_1 so as to maintain clearer visual separation and cover more area. The cohesion vector, $\vec{c}_f$, is computed by taking the average of the vectors from this robot's to each neighbor robot's (x, y) position within r. An alignment vector, $\vec{a}$, is also used only by the flocking behavior, but was computed differently depending on whether or not the robot was a leader. If the robot is a leader, $\vec{a}$ is set to match the control input (heading) given by the operator. If the robot is not a leader, but it is within range r of one, this robot will set $\vec{a}$ to match the closest leader. If the robot is not a leader nor in range of one, $\vec{a} = \sum_{n=1}^{N} a_n$, where a_n represents the alignment vector of the n-th neighbor in N, the set of neighbors of this robot within range r. In other words, a robot's alignment vector is equal to the average of its neighbors' alignment vectors.

If the robot is a leader, the goal vector is represented by the following (notice that leaders do not have a repel vector):

$$\vec{g} = w_a\vec{a} + w_c\vec{c}_f \tag{3.2}$$

If the robot is not a leader, the goal vector is represented by the following:

$$\vec{g} = w_r\vec{r}_f + w_a\vec{a} + w_c\vec{c}_f \tag{3.3}$$

3.1.4 Movement Towards Goal

In each the above behaviors, both velocity $v = ||\vec{g}||$ and angular velocity, α are capped at v_{max} and α_{max}, respectively. Here, α is the difference in heading between the goal vector at the previous time step, $\vec{g}_{t-1}$, and the goal vector at the current time step, $\vec{g}_t$. Once a

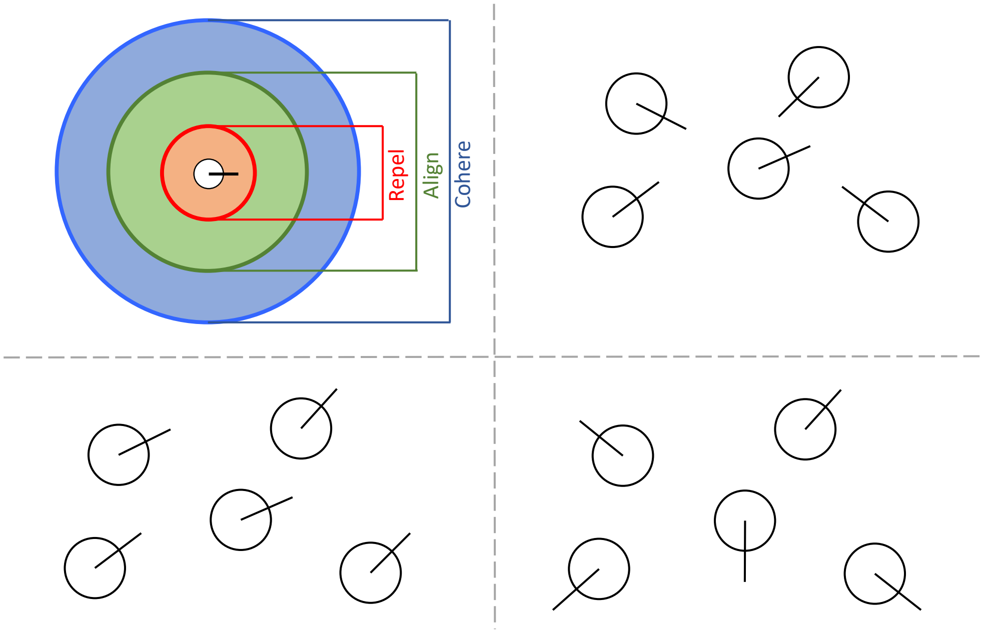

Figure 3.1: The three behaviors used in the presented studies: rendezvous (top right), flocking (bottom left), and dispersion (bottom right). The diagram in the top left shows the three regions used in generating the behaviors.

robot has computed its goal vector using the relevant component vectors for the current behavior (repulsion, flocking, cohesion), the next state of the robot is computed by first turning the robot toward the heading of the goal vector, up to a maximum change of α_{max}, the maximum angular velocity. Because there were 60 simulation steps per second, for each step this maximum angular velocity would be $\alpha_{max}/60s$. Once rotated, the robot would then move forward at the maximum velocity v_{max}. Again, for each step this would be $v_{max}/60s$.

As mentioned previously, the parameters discussed here (e.g. v_{max}) differ between the presented studies and thus will be defined within their corresponding sections. Because the studies involved different types of tasks with different sizes of swarms, it was impractical to use the same parameters for each study.

3.2 ARCHITECTURE OVERVIEW

The simulation is written in CUDA C and uses a significant amount of parallelization with a GPU to perform the computation of each robots control loop in parallel. Due to the large sizes of the swarms in the first study, parallelization is necessary in order to maintain a constant 60 Hz control loop for each robot. CUDA is NVIDIA's free-to-use parallel computing framework usable on the majority of their consumer and business GPUs [89]. OpenGL is used to draw the environment and robots within it, and to handle user input, such as giving new directions to a swarm, when necessary. The simulation consists of both a GPU and CPU step (see Figure 3.2), where the GPU step (performed first) handles leader selection and the control of each individual robot, and the CPU step (performed last) handles the visualizations given in Section 3.4 (except full information), data logging, user input, and some of the computations of global swarm properties.

CUDA allows for interoperability with OpenGL, which is implemented in the simulation by treating each robot's position as a member of an array mapped to an OpenGL vertex buffer object (VBO). During a simulation step, the VBO is mapped to this CUDA array of positions, and the GPU then launches two kernels to handle leader selection and each robot's simulation step in parallel. The use of parallelization allows a significant increase in the number of robots the simulation can handle—if the CPU-side computations are relatively light, up to 2^{14} robots can be simulated at 60 Hz performing any one of the behaviors discussed in Section 3.1. In addition to the positions stored in the VBO, the robots' velocities, leader status, and other variables are stored as arrays on the CUDA device and used in the computation of the robots next position within a simulation step. Due to the VBO mapping, the positions of the robots need not be copied back from GPU memory between each step to be displayed, and instead are directly updated on the screen just as vertices and textures are updated in common 3D graphics applications. This provides significant advantages, as memory copies between the device (GPU) and host (CPU), especially in frequent, small bursts, is costly. Some non-positional data is copied to the CPU after each step, largely for logging purposes; however, this can be done asynchronously so long as it is not needed for displaying extra elements of the visualization in the current step.

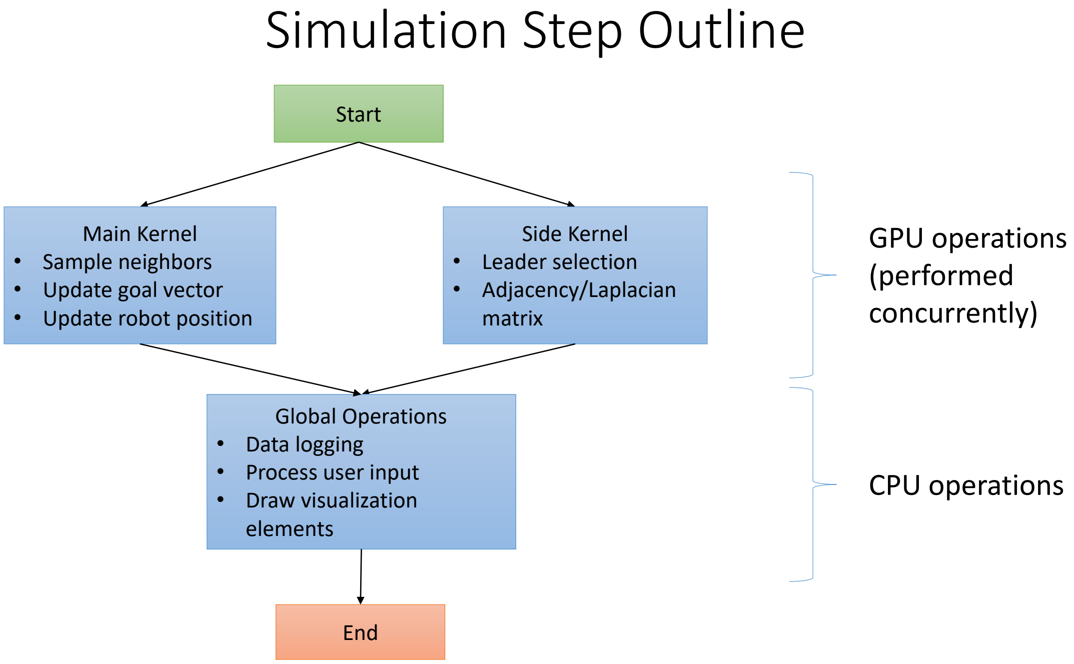

Figure 3.2: An overview of a single simulation step. The simulation first launches concurrent kernels on the GPU to perform leader selection, data collection, neighbor sensing, and position updating. Then the CPU finishes the step by adding extra visualization elements, handling user input, and logging relevant data.

The first CUDA kernel launched by the simulation is reserved exclusively for leader selection algorithms (see Section 3.3 for more details), as well as computation of the adjacency and laplacian matricies of the communication graph of the swarm, which are used for data logging purposes. The second kernel performs the main control loop step for the robots. Within this kernel, a single robot performs all of the operations necessary to update their position for that step of the simulation in parallel with every robot of the swarm. This begins by checking every other robot in the swarm to determine if they are a neighbor, and then updates their *align*, *cohere*, and *repel* vectors according to the active control law (see Section 3.1). The kernel completes by summing these component vectors of the control laws for each neighbor interaction, adding in the obstacle avoidance vector computed from the occupancy grid of the environment (if obstacles are present), and resizing the resulting vector if necessary to ensure its magnitude is no greater than the maximum speed per step ($v_{max}/60$). The angular velocity (angle between the current and previous steps' velocity vectors) is similarly capped at $\alpha_{max}/60$. The robots position is then updated in accordance with this final velocity.

Due to the fact that each robot's control loop and part of the distributed leader selection algorithms is done on its own in parallel, this provides an excellent way to ensure the simulation still maintains the "swarmish" aspect of a robot swarm. Specifically, in enforces the restriction against using centralized algorithms from performing computations that would not be possible in a real-world swarm. For instance, selecting leaders for swarm control could be done at a central point easily and more quickly than a distributed algorithm; however, because there is typically no "central point" of an engineered swarm, this must be done in a distributed fashion. This also provides better scalability and a potentially easier avenue for moving the code to a real robot. Although certainly a simulation is crude approximation for a real-world swarm, the fact that the algorithm used by each member to update its leader status and determine how it will move is independent of other robots means that the underlying architecture of the CUDA kernel can be helpful in programming a real robot performing similar swarming tasks.

3.3 LEADER SELECTION

The following section will detail the two leader selection methods implemented in the user studies. The first of these two, modified random competition clustering (RCC), is adapted from the RCC algorithm in [141] used to select cluster heads for wireless sensor networks in a distributed manner. The RCC method therefore selects leaders that are relatively evenly distributed throughout that swarm, in such a manner where typically no robot is greater than one connection of the communication graph from a leader. In reality, due to the motion of the swarm there are occasionally robots not connected to a leader in between cycles of the algorithm's timers. The second of the two, minimum volume enclosing ellipse (MVEE), is first presented in [71], similarly uses the communication graph of the swarm and selects the robots which, as points, define the smallest ellipse that encloses the entire swarm. In contrast to the RCC leader selection method, MVEE selects leaders only on the perimeter of the swarm. Another important difference is that the number of leaders selected by the MVEE method is constant (eight), whereas the number of leaders selected by RCC varies depending on the structure of the communication network, from a minimum of one to a maximum of $N/2$, where N is the total number of robots.

At this point it is important to introduce the difference between *control* leaders and *information* leaders. For the studies presented herein, control leaders are those that receive the input from the user and then propagate this information throughout the swarm by biasing the control laws of their neighbors. For instance, when the user gives a new goal direction for a flocking swarm, the goal heading is transmitted to the leaders, who then set their alignment to this new direction (see Section 3.1.3), which is then observed by the neighbors of this leader. Information leaders, however, are leaders that aggregate information about the swarm from its members and then relay this information back to the user. Oftentimes control and information leaders will be the same, but this is not necessary. In particular, for the studies in this thesis, the *information* leaders are what will vary between display types, whereas the *control* leaders will always be those selected by the RCC method.

3.3.1 Random Competition Clustering

The first leader selection method I will introduce is the modified version of the Random Competition Clustering (RCC) algorithm for selecting cluster heads in WSNs [141]. This algorithm runs on each robot, and takes as input the period of the leader update timer, t. This timer is the number of steps after which a robot will switch its leader status (some random variation is added by each robot during the algorithm). Pseudocode for the following modified RCC algorithm can be found in [135].

When the algorithm begins, the robot initializes its leader value l to false and its timer $t_0 = t$ plus some variance ϵ. In each of the presented studies, $t = 600$. During each step of the simulation, the robot will first broadcast its leader value to each of its neighbors, and then receive the leader status of each of its neighbors. If at this point the robot's leader timer has expired, it will flip its leader status (i.e., if it is currently a leader it will become a non-leader, and vice versa) and reset its leader timer to $600 + \epsilon$. If the leader timer has not expired however, it will check each of its neighbors' leader statuses to determine if any neighbors are leaders. If so, it will reset the leader timer to prevent two neighbors from becoming leaders at the same time, otherwise, the robot will do nothing. Finally, the robot will decrease the leader step timer by one step, and wait for the next step to begin.

3.3.2 Minimum Volume Enclosing Ellipse

The minimum volume enclosing ellipse algorithm similarly treats the swarm as a communication graph, where robot is a vertex, and an edge between two vertices exists if the two robots are within communication range. The MVEE algorithm works by computing a coreset of up to eight robots in a distributed manner that roughly correspond to the points which define an enclosing ellipse—specifically, the smallest such ellipse that covers each member of the network. Unlike the RCC leader selection algorithm, the MVEE method is unmodified from its original source, the details of which are explained in [71].

3.4 VISUALIZATION METHODS

Four different display types are used to show the current state of the swarm to the participant. The first is called *Full Information*, and shows each robot's (x, y) position and heading updated at 60Hz. This display requires a high amount of bandwidth from the swarm to operator, as it must transmit three variables (x-coordinate, y-coordinate, and heading) for each of the robots at 60Hz.

The second display type, called *Centroid/Ellipse* shows only three pieces of information. First, an ellipse bounding the swarm displayed in red, using the distributed MVEE algorithm presented in [71] to select a set of robots that defined the ellipse. A second smaller ellipse,

Algorithm 1 *modified_RCC (int t)*, where t is the step countdown timer after which this robot will become a leader (600 steps in the user studies presented here).

```
Define int $resetTimer() = \{t + rand(1, 60)\}$
Define leader status $l = false$
Define timer $t_0 = resetTimer()$
while $true$ do
  $broadcast(i, l)$
  $N = \{l_0 ... l_n\} = getNeighborSet()$
  if $t_0 = 0$ then
    if l = false then
      $l = true$
      $t_0 = resetTimer()$
    else
      $l = false$
      $t_0 = resetTimer()$
    end if
  else
    for $l_i \in N$ do
      if $l_i = 0$ then
        $t_0 = t$
      end if
    end for
  end if
  $t_0 = t_0 - 1$
end while
```

in yellow, is displayed within the bounding ellipse. This smaller ellipse bounded the middle 50% of the swarm, which allows the participants to see some measure of spatial distribution throughout the swarm. For instance, if the inner ellipse is off-center relative to the bounding ellipse, the participant can infer the swarm is denser on one side than the other. The final piece of information shown to the participants is a green cross, with the intersection at the centroid of the swarm. The centroid is defined as midpoint of both the minimum and maximum coordinate (x, y) in each dimension. This display requires significantly less information, as the swarm need only transmit the eight points which define each ellipse (32 variables), plus two variables for the coordinates of the centroid.

The third and fourth display types show participants a small subset of the swarm, called the *leaders*. The first of these display types show only the leader set computed by MVEE algorithm, which is referred to as the *MVEE Leaders* display. This requires the lowest amount of bandwidth amongst all display types, as the algorithm gives a maximum of eight leaders, which can be defined by 16 variables (x coordinate and y coordinate for each). The second display type shows the leader set as computed by the modified random competition clustering (RCC) algorithm, first presented in [135] and based on the RCC algorithm in [141]. This is referred to as the *RCC Leaders* display. The amount of information to be transmitted using this display type depends on the number of leaders selected, which in turn depends on the structure of the communication graph of the swarm. The number of leaders can range from one (if the graph is a star shape, with the leader in the middle and all other robots connected to it) to $N/2$ (if the communication graph forms a chain). All display types are shown visually in Figure 3.3.

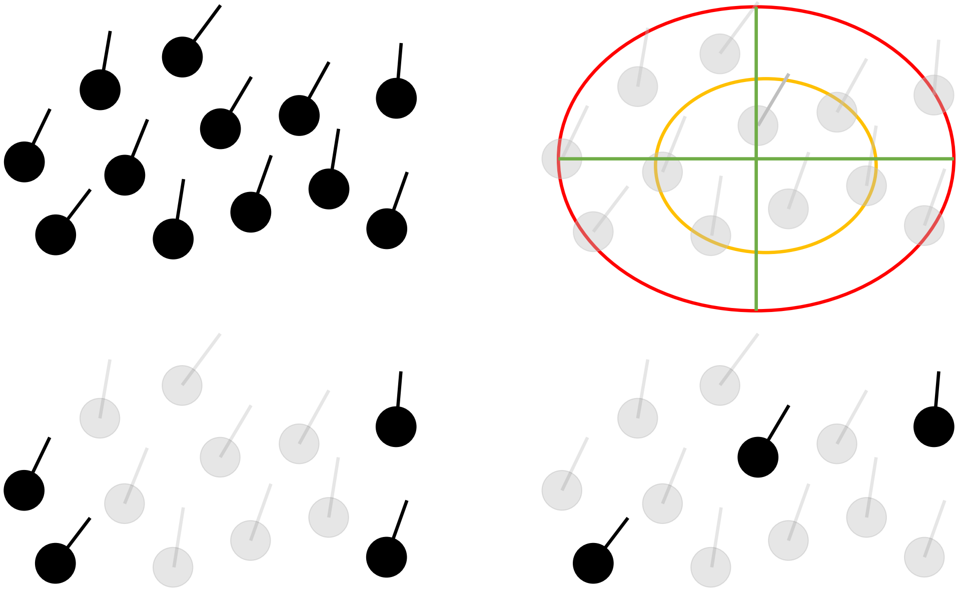

Figure 3.3: The four display types used in the presented studies: Full Information (top left), Centroid/Ellipse (top right), MVEE Leaders (bottom left), and RCC Leaders (bottom right). Greyed-out robots are still present in the simulation, but not visible to the participants.

4.0 USER STUDIES

As mentioned previously, my prior studies present a broad overview of the questions I will focus on in the proposed dissertation. The first studies investigating the effect of communications restrictions on an operator's ability to control a flocking swarm introduced the concept of neglect benevolence, and showed that limits in the communication channel could be overcome by smart displays and intelligent use of information [138, 90]. The leader-based control studies demonstrated that swarms could be controlled using dynamically selected leaders, which allows the operator to send commands to only a small subset of the swarm, instead of each robot [136, 134, 135]. This will likely be a common method for information aggregation to facilitate visualization. Together, these studies show that HSI can be greatly improved by focusing on the aspects of the swarm state that best aid operators in recognizing, predicting, and controlling a swarm.

The studies presented in this chapter will address the three questions posed earlier. The first serves as a proof of concept that different emergent behaviors are recognized differently, and thus may require different levels of information to be shown (Section 4.1). The second user study looks at how visualization can aid in recognition of the swarm's trajectory or goal state (Section 4.2). For many behaviors, this will entail a recognition of consensus. For instance, for flocking, the swarm should eventually reach consensus by moving along a physical trajectory towards a goal direction. Specifically, this study will investigate how different proposed visualizations can aid in recognition of goal trajectory for different swarm behaviors. The third study will investigate the performance of the different visualizations used in a control task, where operators are asked to move a swarm between multiple goals regions, and hopefully demonstrate that proper visualization is important and that full information about the state can be summarized to improve bandwidth requirements without sacrificing performance (Section 4.3). The final study will investigate how visualizations impact user trust in a swarm performing a target-searching task, and demonstrate that more information about a swarm's state and progress will improve both performance and trust (Section 4.4).

4.1 FIRST STUDY: BEHAVIOR RECOGNITION

The study involving participants dealing with a system that exhibited neglect benevolence [86] showed that operators could learn to better time their inputs over the course of interacting with the swarm. Because having a good internal model of a system—and recognizing how the system evolves—usually helps with deciding and timing inputs, this naturally begs the question of how well an operator can recognize these emergent behaviors in general. This is especially important if the behaviors take some time to reach consensus. Furthermore, if we are able to understand which behaviors are easier or harder than others to recognize, this could help designers of interfaces for human-swarm systems know what properties of the swarm state to display, and how much information is needed for the operator to best learn to time their inputs during different exhibited behaviors.

To that end, this study is designed to investigate how well operators recognized three of the most common emergent swarm behaviors as an investigation into how we might improve human control of swarms in general. This study contrasts and complements [116], which studied perception of biological swarm behaviors and subsequently compared their perception with that of simpler displays of rotating dots. Similar to the aided condition [86], the display used in this study shows robot positions and velocities to inform the operator about the current state of the swarm. This study asks participants to discriminate between three types of behavior in the presence of background noise: rendezvous, flocking, and dispersion.

4.1.1 Study Design

This study involves the human participant viewing a series of videos with one of three types of swarm behaviors: rendezvous, where each robot moves to the center of the bounding x- and y-axis values of its neighbors (the parallel circumcenter algorithm [14]); flocking, where each robot tries to match the velocity of its neighbors while also maintaining a minimum distance from close neighbors and maximum distance from far neighbors; and dispersion, where each robot moves away from the average position of its neighbors. The values for the parameters used for these control algorithms (Section 3.1) are given in Table 4.1. The heading input used by the leaders as defined in Section 3.1 is randomly assigned from $\pm\pi$. In each video, there was some amount of background noise, i.e., some of the swarm members moved randomly, ignoring all neighbors. The goal of the study is to determine how much background noise can be present before the participants stop recognizing the behavior being performed. Each participant started at 50% background noise for each behavior. If the participant answered correctly for a behavior, the next time they view a video with that same behavior the noise level was increased according to the following formula:

$$e_1 = (100 + e_0)/2 \tag{4.1}$$

Where e_0 and e_1 are the current and new noise percentages, respectively. In other words, the noise increased to the halfway point between the previous noise level and 100% noise. If the participant answered incorrectly, the next time they viewed that behavior the noise level would be set as follows:

$$e_1 = (50 + e_0)/2 \tag{4.2}$$

Note that this means the noise level can never fall below 50%, even if a participant responded to a video with 50% noise with the incorrect behavior. In this study, participants always correctly recognized a behavior with 50% background noise. The participants viewed each behavior six times, along with six videos with 100% noise, for a total of 24 videos. The videos with 100% noise serve as a baseline to ensure that participants can also discriminate between an organized behavior and no behavior at all. All videos are presented in a random order.

Table 4.1: Simulation parameter values for recognition study.

Variable	**Value**
d_1	0.5 meters
d_2	1.5 meters
r	2.0 meters
v_{max}	1.0 meters/sec
α_{max}	6π radians/sec
w_a	1.0
w_c	0.9
w_r	1.0

The videos are generated via the simulator described in Section 3.2 with 2,048 robots. The robots begin at random positions in a 2D plane, bounded by [-15,15] meters in both the x- and y-axis. The viewport shows as far as ±20 meters in each direction, to allow the participants to see the entire swarm at the beginning, as well as to allow for some expansion in the overall swarm area before leaving the viewport (especially important for the dispersion behavior). The videos are each 20 seconds long, to give the participants plenty of time to view the behaviors and distinguish between them, although participants could select a response at any time. Total viewing time before selecting a response is recorded along with their response for data analysis. A simulation step is performed 60 times a second, whereby each robot sensed its neighbors and moved according to the algorithms described in Section 3.1, giving a total of 1,200 steps per video.

4.1.2 Participant Details

Participants were selected from the Amazon Mechanical Turk user-base, and were given a short questionnaire to complete, asking their age, gender, average weekly computer use, and average weekly spent playing computer games. After the videos were finished, each participant was asked to describe their strategy for recognizing the behaviors, if any. A total of 72 participants were collected. Of these, 32 participants were female and 40 were male, and ages ranged from 20 to 72 years old (median of 32).

4.1.3 Hypotheses

Because of humans' innate ability to recognize biological motion and common fate [55, 123], the first hypothesis is that participants in general will be good at recognizing all behaviors, even when background noise is high. Results in [123] report discrimination with noise levels as high as 97%, although because these behaviors are more complicated than simple straight motion in a common direction, it is expected here that noise thresholds for recognition will be slightly lower. More importantly, the second hypothesis is that there will be significant differences between the behaviors in terms of recognizably, and in the factors, including individual differences, that give rise to recognizability. Furthermore, it is believed that the flocking behavior will benefit from longer viewing times before a response is given due to the time it takes for consensus to emerge and the collective movement to begin before the behavior becomes apparent. To summarize, the hypotheses for this study are as follows:

- **H1**: The average maximum level of background noise with correct recognition will be high ($> 80\%$).
- **H2**: There will be significant differences between the emergent behaviors in the average maximum level of background noise with correct recognition.
- **H3**: Flocking will benefit from longer viewing time before response, whereas rendezvous and dispersion will not.

4.1.4 Results

Due to how the noise was adjusted for each behavior-type (see Section 4.1.1), the final correct answer given by the participant for each behavior is also the maximum noise percentage for that behavior where recognition was still successful. Therefore, for each participant, we can easily find the maximum noise percentage with correct recognition for each behavior. Using a Welch's t-test, rendezvous was found to be significantly easier to recognize than either

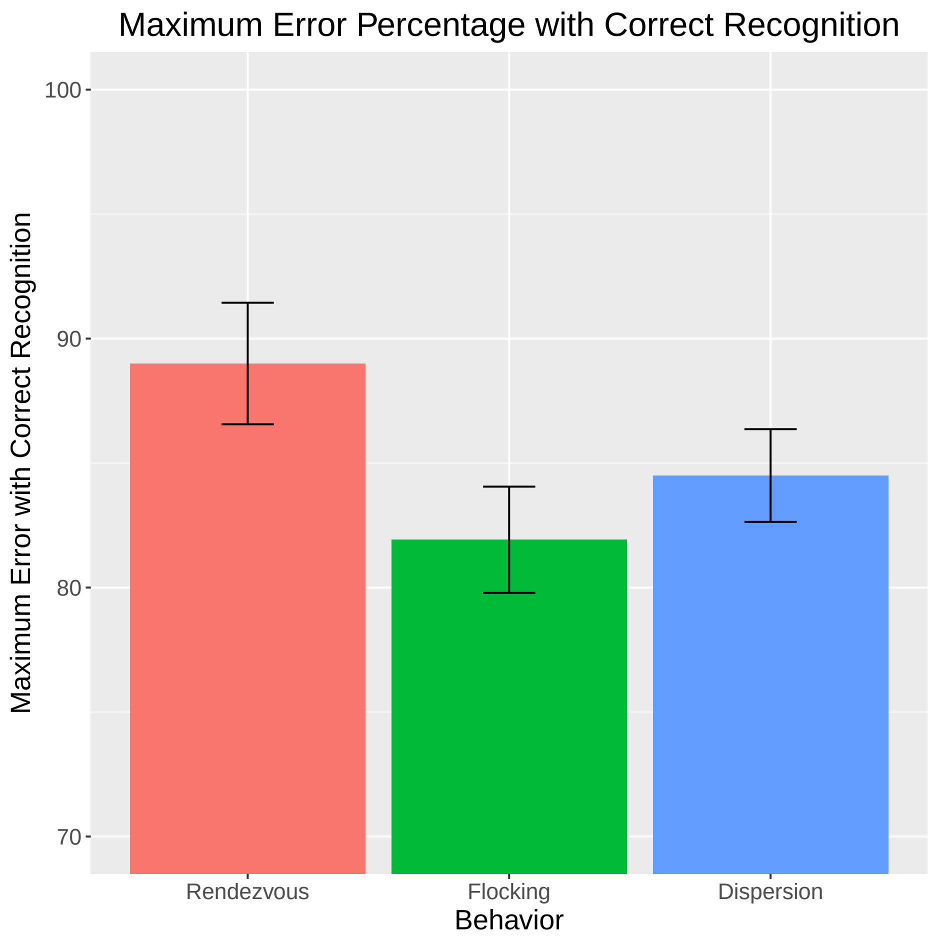

Figure 4.1: The average highest percentage of background noise at which participants still recognized the emergent behavior correctly.

flocking ($t(137.86) = 3.521$, $p < .001$) or dispersion ($t(140.16) = 2.619$, $p = .01$). Flocking and dispersion were not significantly different ($t(141.49) = 0.889$, $p = .375$). Figure 4.1 shows these results.

A similar measure of recognizability is the number of correct answers for each behavior, which should roughly correlate with the maximum noise percentage presented previously. Indeed, the same results were found here. Using a Welch's t-test, rendezvous allowed for more correct answers by participants than either flocking ($t = 3.153$, $p = .002$) or dispersion ($t = 4.146$, $p < .001$), with no significant difference between flocking and dispersion ($t =$

1.278, $p = .204$). Interestingly, results show that participants correctly recognized a lack of behavior (100% noise) significantly less often than any of the three behaviors ($t = 5.310$, $p < .001$ for rendezvous; $t = 3.480$, $p < .001$ for flocking; and $t = 2.645$, $p = .009$ for dispersion).

Results also show that, on average, the longer a participant views a video before submitting a response the more likely they were to be correct, but only for the flocking behavior (see Figure 4.2). Taking longer to view a video of flocking behavior before giving a response corresponded to correct responses at higher noise percentages for flocking ($F = 14.94$, $p < .001$, $r^2 = 0.164$). This effect was marginally significant for rendezvous ($p = .053$), and there was no such effect for dispersion ($p = .919$). These results were again mirrored when considering total correct answers instead of maximum noise.

Effects of demographic data were also analyzed to determine if age, gender, computer use, or video gaming frequency impacting the performance of operators. While there was no correlation between age and the average maximum noise across all behaviors and responses, age did have a small positive correlation with total average viewing time of the videos ($F(1, 70) = 5.697$, $p = .020$, $r^2 = 0.062$), although this did not translate to higher maximum noise values for correct responses in the flocking behavior as might be expected. Surprisingly, there was a difference in performance between genders, with females recognizing behaviors at a higher noise level than males ($F(1, 70) = 5.26$, $p = .025$), although like the correlation between age and viewing time, this effect was small. This could be due to the fact that females, on average, viewed the videos for a longer period ($F(1, 70) = 2.975$, $p = .089$, $r^2 = 0.027$), although this effect was also marginal. Computer usage was assessed by asking participants to estimate how often they used a computer in a week, at 10-hour intervals (i.e. 0-10 hours, 10-20 hours, etc.). Higher computer use correlated with better recognition at higher noise rates, but for rendezvous only, and the results were only marginally significant ($F(1, 70) = 3.00$, $p = .088$, $r^2 = 0.027$).

Perhaps the most telling results for individual differences come from the subjective qualitative responses of the participants. The final question of the survey asked if they could describe any strategies they used for recognizing behaviors, and there were many common themes across the responses, primarily supporting the idea that humans are good at recog-

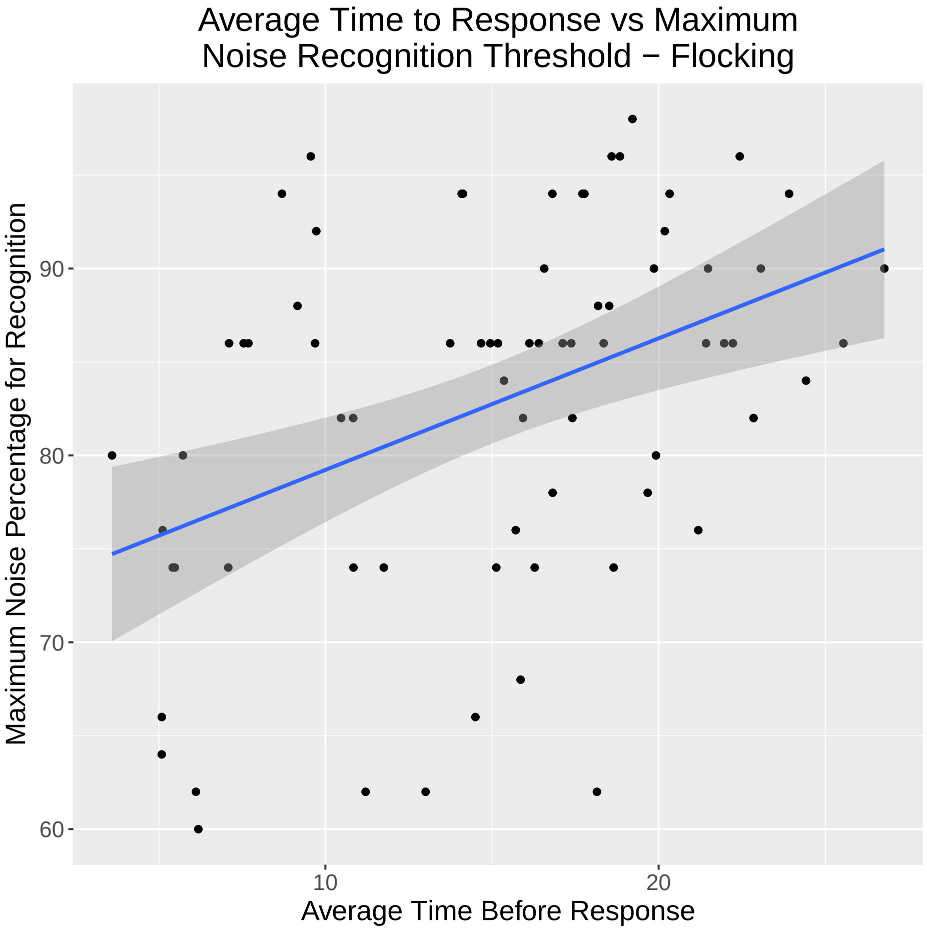

Figure 4.2: The correlation between time to issue a response and highest correct recognition threshold for the flocking behavior.

nizing patterns and collective motion. A common strategy seemed to be to unfocus and view the bigger picture to recognize global patters, instead of focusing on individual robots. For instance, many participants mentioned “unfocusing [their] eyes” or “watching for patterns to emerge”. Some characteristic responses are reported in Table 4.2.

4.1.5 Discussion

The results of this study and feedback from the participants clearly show that some behaviors are easier to recognize than others, confirming **H2**, and provides evidence that humans

Table 4.2: Behavior recognition strategies reported by participants in recognition study.

ID	Description of Strategy
p8	“I tried to look at everything as a whole and pick out certain patterns”
p15	“I tried to unfocus my eyes and recognize the general movement pattern of the dots.”
p18	“I watched for the density to change in the picture and then tried to discern some sort of pattern from that.”
p21	“I tried to let my focus widen and not stare too hard.”
p30	“The strategy I used for recognizing behaviors was to unfocus my eyes from any particular spot and try to notice if there seemed to be a pattern within the large group.”

use the Gestalt properties of swarm behaviors such as common alignment, common velocity, and proximity to recognize different behaviors. Rendezvous, which involves an easily visible aggregation to a common point, was the easiest to recognize from background noise. Furthermore, flocking benefited from longer viewing times, as this gave the user more time to pick out the common fate of the flocking (non-noise) swarm members, which confirms **H3**. The average highest level of noise where recognition was still successful is 85.38%, which as hypothesized is high, but not as high as the signal-to-noise ratio for common fate reported in [123], confirming **H1**. This is likely because the swarm behaviors tested were slightly more complex than simple common motion—primarily due to interactions with neighbors.

The responses by participants further reinforce the idea that operators take a holistic approach to viewing the collective motion inherent in emergent swarm behaviors. This could mean that the underlying metaphor used for the design of swarm algorithms may be less important, as long as the end result is recognizable via base perceptual mechanisms, such as collective motion and common fate as studied in [123]. Furthermore, this also helps explain why flocking was the only behavior to benefit from longer viewing times. Flocking requires a period of time where the robots have not yet reached a consensus on direction, whereas rendezvous and dispersion do not. These results lay the groundwork for the future studies, which will use the different display methods presented in Section 3.4 to determine how the visualization of a swarm can be simplified to save on both bandwidth and visual complexity while still maintaining the ability for operators to recognize and control a swarm.

In light of the recent results presented here, in [116], and the results of work on neglect benevolence, the next logical step for this line of research is to investigate how different visualizations of a swarm affect users' abilities to predict future states of the swarm. For instance, are there ways of displaying summary information about a swarm (namely, centroid and spatial bounds or a subset of leaders) that improve or maintain predictability when compared to viewing full information (i.e. each individual robot's position and heading)? Because summarizing this information can be done at the swarm level, it necessarily requires a lower amount of bandwidth to transmit this data between a swarm and remote human operator compared to full information.

4.2 SECOND STUDY: BEHAVIOR PREDICTION

The goal of this study is to determine if different methods of displaying a swarm change the ability of an operator to predict the swarm's future state. During a single trial, participants were shown a swarm performing one of three different behaviors: rendezvous, flocking, or dispersion. Furthermore, the swarm is displayed on the screen using one of four display methods (see Section 3.4). Each trial lasted 30 seconds, during which the participant was asked to draw on a touch screen what they predicted to be the final shape of the swarm at the end of the trial. Finally, each participant saw each behavior-display pairing twice, giving a total of 24 trials over the course of the study (3 behaviors × 4 display types × 2 viewings).

Table 4.3: Simulation parameter values for prediction study.

Variable	**Value**
d_1	1.0 meters
d_2	2.0 meters
r	5.0 meters
v_{max}	1.0 meters/sec
α_{max}	6π radians/sec
w_a	1.0
w_c	0.9
w_r	1.0

4.2.1 Study Design

The swarm consisted of 256 robots, which begin at random positions and orientations within a 24 × 24 meter box centered at a random point within 10 meters of the origin of the simulation environment. The environment extended from ±100 meters in the x-dimension, and ±35 meters in the y-dimension. These dimensions were chosen because they gave a close enough view of the swarm to make each detail of the GUI visible, while still allowing the entire swarm to be visible throughout each trial, regardless of the current behavior being performed. The values for the parameters used in the control algorithms (Section 3.1) are given in Table 4.3.

Participants used a 23-inch Dell touch screen display with a resolution of 1920 × 1080p to both view the swarm and give input indicating their prediction. During a trial, the participant is asked to use the touch screen to draw a shape that they believe would bound the final position of the swarm at the end of the 30-second trial period. A progress bar showing the time remaining in the trial was displayed at the top of the screen at all times. Participants could give multiple predictions over the course of the experiment—allowing them to update previous predictions if they deemed them inaccurate. Each prediction input was converted into the convex hull of the user-drawn points and, along with the convex hull of the final positions of the swarm members, is used for data analysis purposes to determine the accuracy of the prediction input.

Before completing the 24 trials in a random order, the participants were allowed to complete three training trials to get accustomed to the interface and touch screen. These training trials consisted of one each of the three behaviors, all with the Full Information display. Upon completing the 24 trials of the main experiment, participants finished with a short survey, asking them to rank the four displays in order of their perceived helpfulness, both overall and for each of the three behaviors. Because each participant viewed all trials in every condition, this study uses a within-subjects design.

4.2.2 Participant Details

Participants were recruited from the University of Pittsburgh and surrounding area. There were 26 participants in total, with an age range of 18-65 years old (median of 25). Of the 26 participants, 13 were female and 13 were male.

4.2.3 Hypotheses

The first hypothesis is that the Full Information and Centroid/Ellipse display types will give the best prediction accuracy across all behaviors, and will be equal to each other in terms of accuracy, similarly to what was found in [90]. Therefore, the hypothesis also states that the leader-based displays will both give lower predictions than either the Full Information or Centroid/Ellipse displays overall; however, for individual behaviors that may not hold true. Specifically, for the flocking behavior, where the current heading of the swarm members is necessary for determining their future position, the hypothesis is that one or both of the leader-based displays will outperform the Centroid/Ellipse display, as the latter does not display any heading information. Finally, the second hypothesis is that rendezvous will give the highest prediction accuracy overall, as it was shown to be easiest to recognize in the previous study. To summarize, the hypotheses for this study are as follows:

- **H1**: The Full Information and Centroid/Ellipse displays will give the best prediction accuracy overall, but the leader-based methods will be best for flocking specifically.
- **H2**: The rendezvous behavior will give the highest prediction accuracy overall.

4.2.4 Results

The main method for evaluation of participants' predictions of the final state of the swarm was accuracy as compared to the actual final state (the positions of the robots at the end of the 30 seconds of the trial). Each time the user gave a prediction, their chain of drawn points was used to compute a convex hull around those points. Similarly, the final robot state was represented by the convex hull of the positions of the robots at the end of 30 seconds. Given two convex hulls, it is therefore easy to compute the accuracy according to the following equation:

$$A = (H_u \cap H_r)/(H_u \cup H_r) \tag{4.3}$$

where A is the accuracy variable, H_u is the convex hull of the user-drawn estimate, and H_r is the convex hull of the final robot positions. This measure of accuracy was used because it penalizes inaccuracy in the position of the swarm, as well as both overestimating the shape (i.e. drawing a large shape covering much more area than the swarm takes up in reality) and underestimating the shape (drawing a small shape in the middle of the swarm).

Using an analysis of variance test (ANOVA), results presented in Figure 4.3 show that there are significant differences for accuracy between both behavior ($F(2, 1651) = 414.4$, $p < .001$) and display type ($F(3, 1650) = 8.68$, $p < .001$). The accuracy of predictions for the dispersion (65.8%) behavior were significantly higher than both rendezvous (43.1%, $t = 26.97$, $p < .001$) and flocking (40.0%, $t = 27.63$, $p < .001$). Accuracy for rendezvous was also significantly higher than flocking ($t = 3.35$, $p < .001$). For display types, there was no significant difference between the accuracies for the Full Information (50.5%) and Centroid/Ellipse (51.2%) displays; however, both were significantly more accurate overall than the MVEE (45.5%, $p < .001$ for each) or RCC (46.9%, $p = .008$ and $p = .002$, respectively) Leader displays. Accuracy between the two leader-based displays was not significantly different.

As one of the main hypotheses is that different behaviors will require different displays to maximize the accuracy of prediction, the data within each behavior were analyzed to determine if this hypothesis is true. Results show that, while accuracies in the two leader-based displays were lower than either the Full Information or Centroid/Ellipse displays overall across all behaviors, when investigating the dispersion behavior significantly prediction accuracy using RCC Leaders (65.3%) were not significantly different than the Full Information or Centroid/Ellipse displays, while the MVEE Leaders display gave significantly lower prediction accuracies (62.8%) than both the Full Information ($t(234.11) = 2.88$, $p = .004$) and Centroid/Ellipse displays ($t(238.98) = 3.21$, $p = .002$).

4.2.4.1 Timing of Predictions Accuracy is only one of the important qualities of a prediction; the other is the time a prediction is made. Earlier predictions are more beneficial than later predictions, assuming equal accuracy, because they allow information to be gained earlier. Therefore, the analysis was repeated using only predictions made in the first half (first 15 seconds) of a trial. Results show that the differences in accuracy between each behavior for early predictions remain the same as when comparing between behaviors using all predictions ($F(2, 720) = 522.00$, $p < .001$), with dispersion giving significantly higher accuracy (63.1%) than both rendezvous (34.4%, $t(485) = 25.33$, $p < .001$) and flocking (25.4%, $t(454.14) = 30.62$, $p < .001$). Similarly, early predictions for the rendezvous trials were on average more accurate than those in flocking trials ($t(472.06) = 7.12$, $p < .001$).

When restricting predictions to only the first half of a trial, the differences between the display types overall are not present ($F(3, 719) = 0.91$, $p = .436$). However, when looking within behaviors the differences become evident. For rendezvous, early predictions using the Centroid/Ellipse display were significantly more accurate (37.5%) than the RCC Leader display (31.7%, $t(120.86) = 2.40$, $p = .018$), and marginally significantly more accurate than the MVEE Leader display (33.0%, $t(119.89) = 1.85$, $p = .067$. The Full Information display, surprisingly, was not significantly different from either of the two leader-based displays or the Centroid/Ellipse display.

Similar results are found for dispersion, except with the Full Information display being superior, instead of the Centroid/Ellipse display. Early predictions using the Full Information

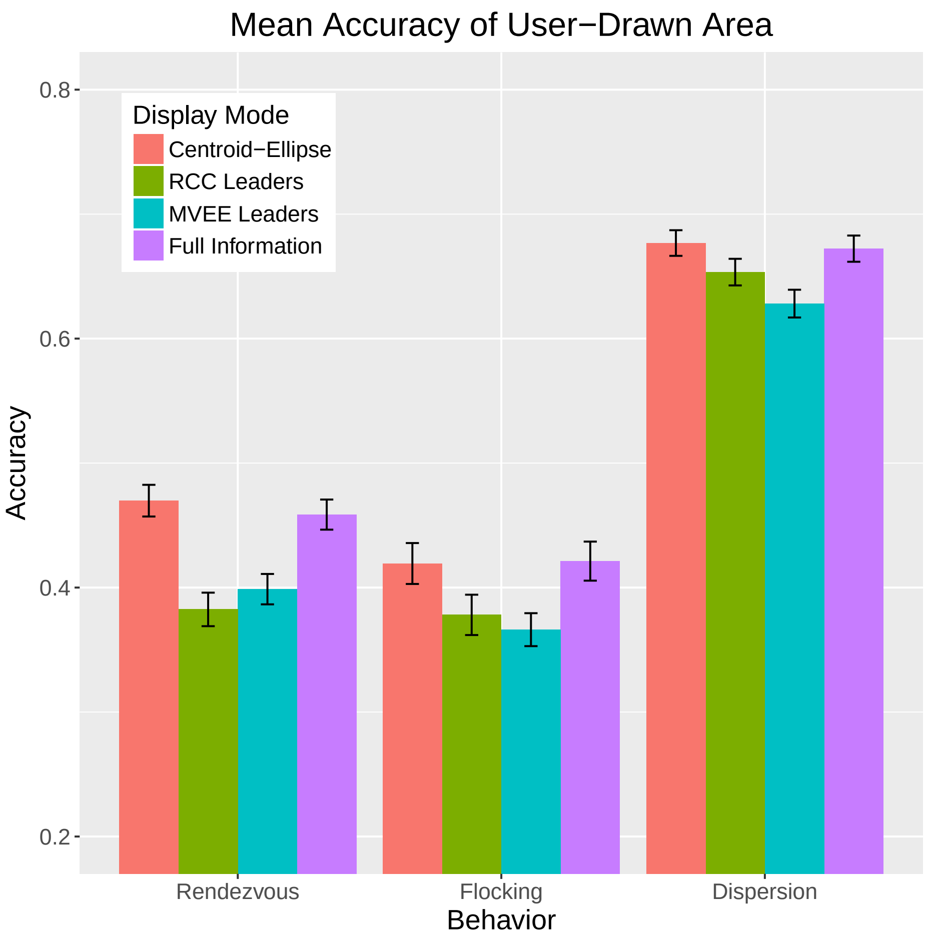

Figure 4.3: The average accuracy of participants' final swarm position estimates across both behavior and display type.

display (66.3%) were significantly more accurate than both the MVEE Leader display (61.6%, $t(90.84) = 2.10$, $p = .038$) and RCC Leader display (61.2%, $t(114.27) = 2.52$, $p = .013$). There were no significant differences between the Centroid/Ellipse display and either leader-based method, nor was there a significant difference in early prediction accuracies between the two leader displays. Overall results of accuracy based on trial time of position estimates during each behavior are available for rendezvous, flocking, and dispersion in Figures 4.4, 4.5, and 4.6, respectively.

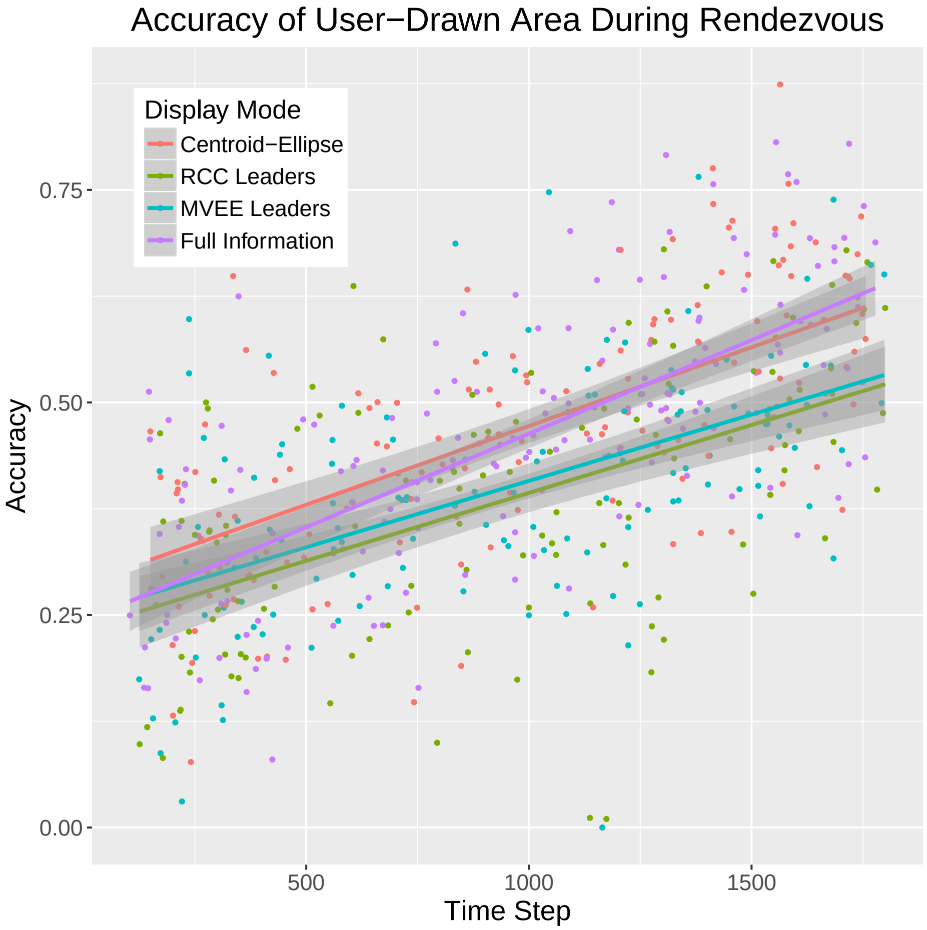

Figure 4.4: Each final position estimate given by participants during the rendezvous behavior, based on the time the estimate was given.

4.2.4.2 Participant Feedback Feedback from participants, while subjective, is important for interface designers of any system merging both autonomy and human intelligence—HSI systems are no different. To get qualitative feedback from participants, they were asked to complete a short survey ranking their preferred display types after the experiment completed. The responses here clearly show that the Full Information condition was widely preferred by participants, with the Centroid/Ellipse display coming in a distant second. Neither of the leader-based display methods received the top ranking overall by any participant; however, the MVEE Leader display does seem to be preferred slightly more than the

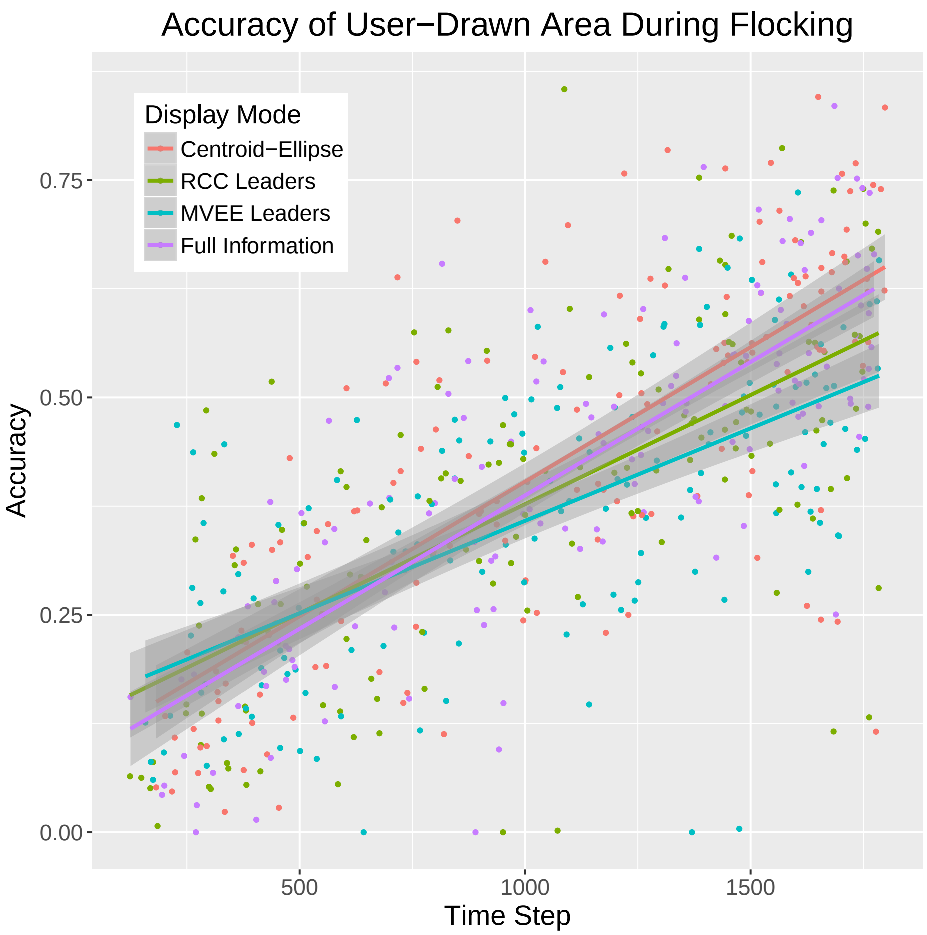

Figure 4.5: Each final position estimate given by participants during the flocking behavior, based on the time the estimate was given.

RCC Leader display (see Figure 4.7).

4.2.5 Discussion

The results of this study suggest that there are differences in the predictive abilities of operators depending on what behavior a swarm is exhibiting, as well as what display mode is being used. While results show that the dispersion behavior allowed for significantly more accurate predictions overall, which allows us to reject **H2**, this could be due to the fact that small differences between the boundaries of the user-drawn and final robot convex hulls make

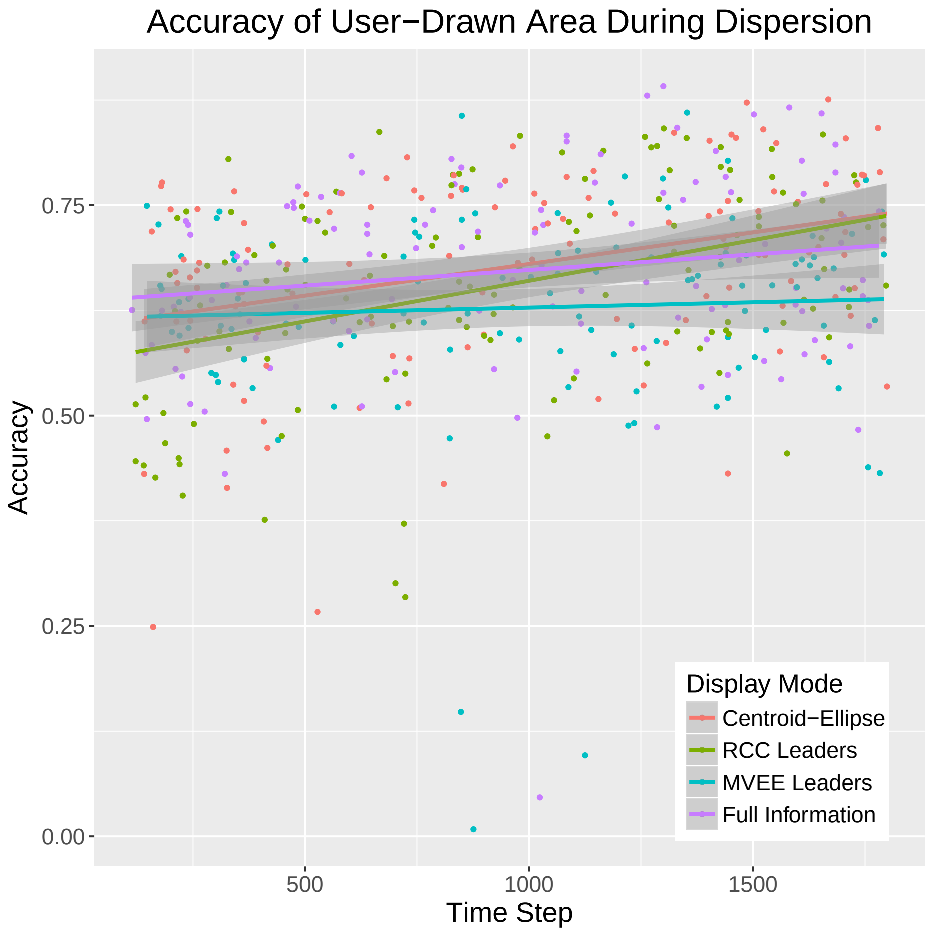

Figure 4.6: Each final position estimate given by participants during the disperse behavior, based on the time the estimate was given.

up a smaller percentage of the overall area of their union than in the flocking or rendezvous behaviors. Specifically, due to the fact the participants could not be perfectly precise down to the pixel level, there is always going to be some inaccuracy in drawing the estimated final swarm shape on a screen; however, because this inaccuracy does not increase with swarm size, it makes up a larger percentage of the overall swarm area when the space covered by the swarm is small—as it is in flocking and rendezvous—thus giving those behaviors lower accuracies.

Results within each behavior are more telling. Because the Full Information and Cen-

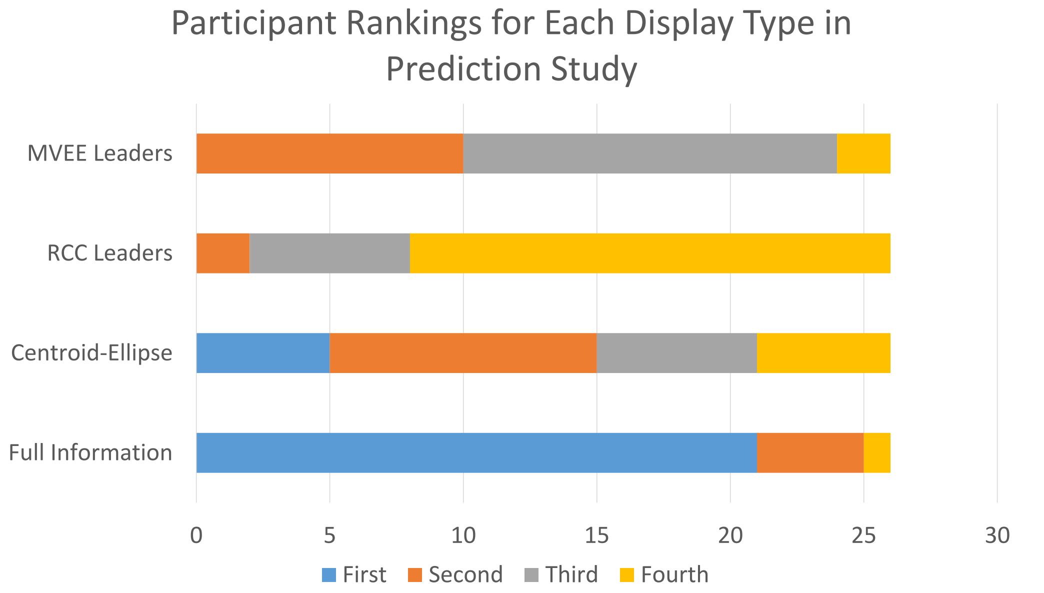

Figure 4.7: Participant rankings of each display type in the prediction study.

troid/Ellipse displays gave equal accuracy regardless of the display type or behavior, in bandwidth-limited situations it may be preferable to display a bounding ellipse and centroid instead of full information about each robot. This confirms the results of the swarm control study conducted in [90], as well as hypothesis **H1**. This does not mean that leader-based displays are useless, however. Because the task in this study involved estimating the spatial qualities of a swarm, the Full Information and Centroid/Ellipse displays may have been superior because they give better spatial information. The Centroid/Ellipse display in particular gives *only* spatial information. The leader-based displays, however, may be better suited for tasks where other characteristics need to be estimated—such as consensus of the swarm on a goal direction, or overall connectivity of the sensing or communication graphs. Future studies could extend this research to predictions where non-spatial information is more important, to see if other display methods become necessary.

Another possible explanation for the high accuracy of predictions in the Centroid/Ellipse display is that this display preserves the global, Gestalt-type properties of a swarm. Namely, with a centroid and ellipse operators get a clear picture of the overall shape and position of the swarm in a single glance. With leader-based displays, operators must infer this information from the positioning of the leaders. Doing so while the swarm is moving may be difficult enough to give the lower performance seen here.

The results of this study lead to a natural next step, which will be covered in the following section: investigating how these different displays for common swarm behavior can be used not only to *predict* swarm behavior, but also *control* it. The next study will investigate how these same displays affect an operator's ability to direct a flocking swarm to goal regions within an obstacle-filled environment. The focus here will be specifically on flocking, because it requires some input from the user to guide the swarm in the proper direction (i.e. the goal heading), whereas rendezvous and dispersion in and of themselves do not require this.

4.3 THIRD STUDY: BEHAVIOR CONTROL

The goal of this study is further test a hypothesis originally stated in the prediction experiment given in Section 4.2: that different behaviors and tasks require the operator to have knowledge of different state variables, and thus the display should be adapted to highlight the most relevant state variables for the task. This experiment expands on the previous by investigating a control scenario—rather than asking the participant to be passive and only give predictions, we are now asking them to actively interact with the swarm.

Similarly to the prediction study, this study investigates how different methods of displaying a swarm affect measures of the operator's performance, this time in a simple task of guiding the swarm past obstacles to a goal region. In this experiment, the swarm operates under a standard flocking algorithm only (rendezvous and dispersion are not used), whereby each robot coheres to neighbors far away, repels from those close, and otherwise tries to align its heading with a nearby leader robot (or, if no leader is available, the average heading of its neighbors). Refer to Section 3.1.3 for details of the flocking algorithm used. Values for the parameters used in flocking algorithm are given in Table 4.4.

Table 4.4: Simulation parameter values for control study.

Variable	**Value**
d_1	0.5 meters
d_2	2.0 meters
r	4.0 meters
v_{max}	2.0 or 1.5 meters/sec
α_{max}	6π radians/sec
w_a	1.0
w_c	1.0 or 1.25
w_r	1.25

4.3.1 Study Design

The swarm consisted of 256 robots, which begin at random positions and orientations within a 24×24 meter box centered at the origin of the simulation environment. The environment extended from ± 200 meters in both the x- and y-dimension. The operator had two possible commands during a trial: change the goal heading of the leaders (done by dragging a line on the screen in the desired direction), or switch between "compact" and "standard" flocking mode. The switch from standard to compact increased the weight of the cohesion vector w_c for each robot from 1.0 to 1.25 and lowers v_{max} from $2.0m/s$ to $1.5m/s$, resulting in a swarm that is more compact and less likely to break apart, but moves slower as a result. The environment always has 20 randomly-generated rectangular obstacles, each with a random length and width between 5 and 50 meters. Overlaps creating more complex, non-rectangular obstacles were allowed, but no obstacles were allowed within the swarm's starting space, or within the goal region.

There are four measures by which participants are evaluated. The main measure is the time (in simulation steps) that it took for them to reach the goal. The goal is always placed at 80 meters away from the center, but due to the random generation of obstacles, the optimal path is not always a straight line from center to goal. Secondary to this is the length of the path the swarm centroid took from the origin (beginning) to the goal. This should correlate with time taken. The two secondary measures are the participants estimate of swarm shape and average heading, which were taken at three randomly-placed pause points for each trial. During a pause point, the simulation stops and participants are asked to draw a shape bounding the swarm (in a manner identical to the prediction study), and then to give an estimate of the swarms average heading (by drawing a line in the estimated direction on the screen).

There are four possible conditions for a trial based on the four display types possible. Each participant completed three blocks of four trials each. Within each block the four possible conditions are given in a random order. Thus, each participant completed 12 trials total, with 3 for each display type, giving 9 pause points per behavior type per participant. Before the participant begins a block, they were allowed to practice controlling the swarm with each display type to become comfortable with the display and controls. Figure 4.8 gives a screenshot of this task under the Full Information condition.

4.3.2 Participant Details

Participants were recruited from the University of Pittsburgh and surrounding area. There were 21 participants in total, with an age range of 19-31 years old (median of 24). Of the 21 participants, 14 were female and 7 were male.

4.3.3 Hypotheses

Given the results of prediction study presented in Section 4.2, the main hypothesis is that the Full Information and Centroid/Ellipse display type will give the performance, as measured by time to reach the goal, across all behaviors. Similarly, the secondary hypothesis is that those same two displays will give the best accuracy for position estimates of the swarm, but that the Centroid/Ellipse display will be the worst for heading estimates, as this display gives no heading information other than what the viewer can infer by watching the centroid move across the screen. In summary, the hypotheses for this study are as follows:

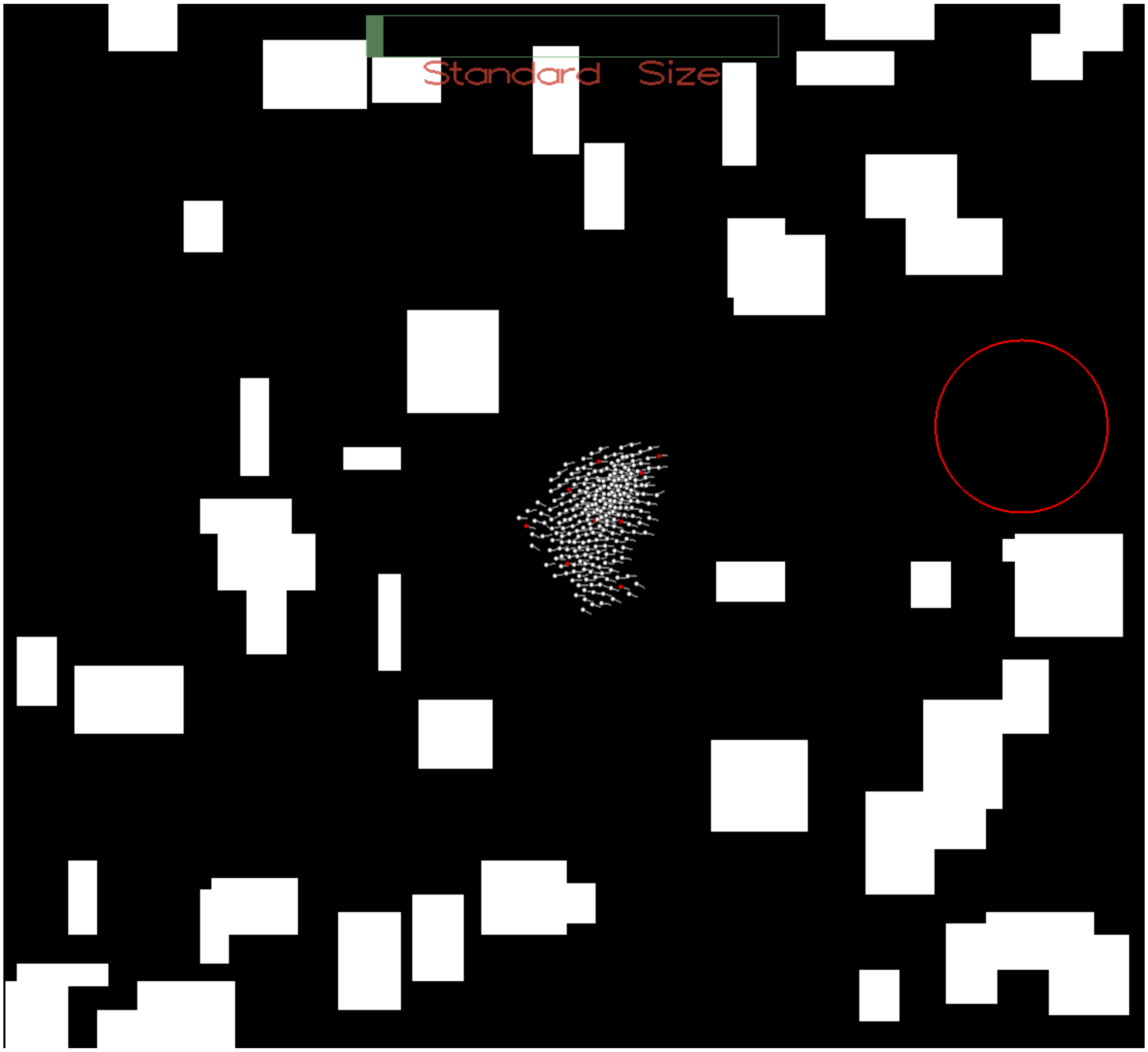

Figure 4.8: A screenshot of the beginning of a trial for the behavior control study with the Full Information display (all robots visible). Obstacles are shown in white and the goal region in red. The top shows a green progress bar indicating how much time remains in the trial, along with text indicating if the swarm was in compact or standard mode. Robots in red are the control leaders (i.e., they respond directly to user input, and bias the goal alignment headings of non-leaders).

- **H1**: The Full Information and Centroid/Ellipse displays will both give the best performance on the main metric (time to goal).

- **H2**: The Full Information and Centroid/Ellipse displays will both give the best performance for the position estimates, yet only Full Information will be best for heading estimates, with the Centroid/Ellipse display being the worst for this measure.

4.3.4 Results

The main result is that there were significant differences between display types for the time it took participants to reach the goal region ($F(3, 233) = 6.165$, $p < .001$) with the MVEE display being significantly worse (longer time-to-goal) than any other method. The other three methods were not significantly different from each other (see Figure 4.9). Next, the analysis looked at the quality of the position estimates given by participants. As with the previous study, accuracy is given by equation 4.3. The results again show a difference between display types for position estimate accuracy ($F(3, 233) = 12.1$, $p < .001$). Further analysis shows accuracy of user estimates were statistically identical for the Full Information and Centroid/Ellipse display modes, which matches the performance seen in the previous study. These two modes were both significantly better than the RCC Leader mode ($p < .001$ for each) and MVEE Leader mode ($p < .001$ for each). The MVEE and RCC modes were not statistically different from each other, however (see Figure 4.10).

Results for the heading estimates given by participants when queried similarly show a significant effect of display type on the accuracy of heading estimates ($F(3, 233) = 7.05$, $p < .001$, see Figure 4.11). Results show that Full Information gave significantly lower estimate errors than any of the other displays, while the MVEE Leader display did significantly worse. The RCC Leader display and centroid ellipse display were statistically equivalent ($t = 0.36$, $p = .721$).

Two other secondary measures that were logged were the time taken to give input for both heading and position, and the total length of the path taken by the swarm from initial state to goal. The input time showed no significant differences between display types for either the position or heading input—participants were relatively uniform in how long it took them to give estimates. However, the path length did show significance between the

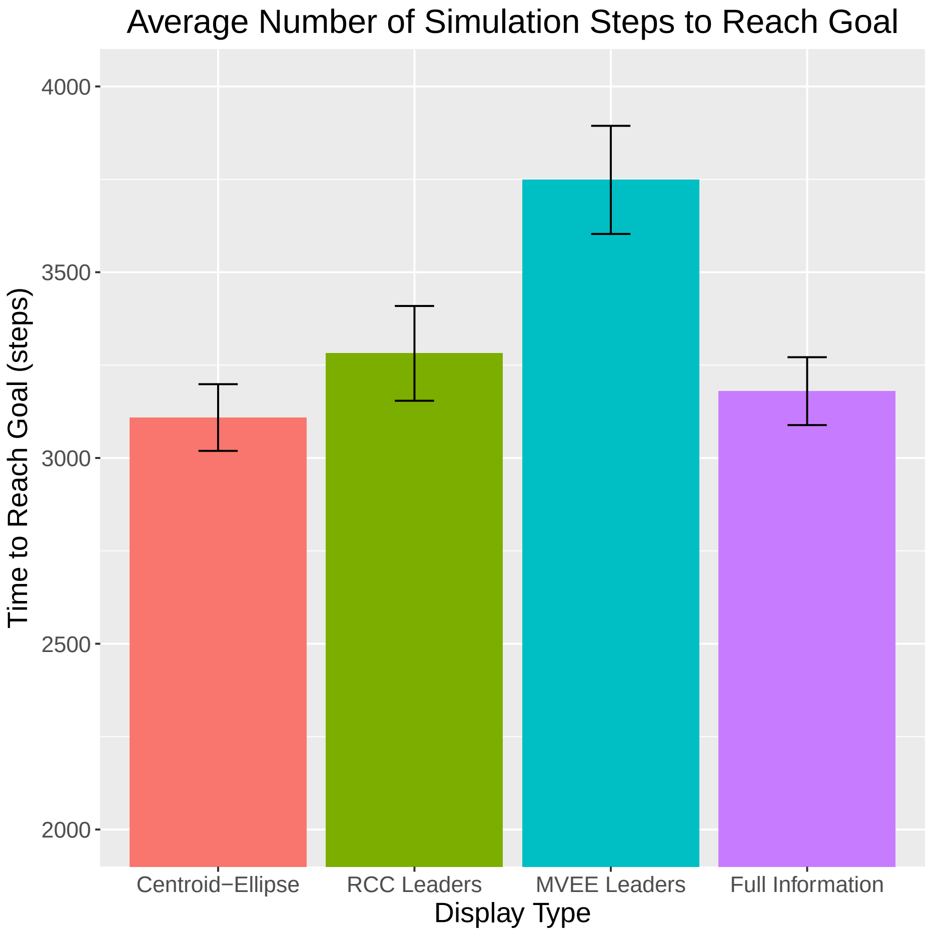

Figure 4.9: The average number of steps to reach the goal region for each display type. Each step is 1/60 of a second (i.e. 1 minute is 3600 steps).

conditions ($F(3, 233) = 2.449$, $p = .063$). The results here match the main results (time to goal), with MVEE Leader conditions resulting in significantly longer paths than any other condition (see Figure 4.12); however, the variance was much higher than that of the time taken to reach the goal. Further analysis shows a very weak correlation between the accuracy of the position and heading estimates and total time to reach the goal, with higher accuracy in both correlating with faster success ($r^2 = 0.05$, $p < .001$ for both).

Finally, Figure 4.13 shows the participant rankings for each display type. They were asked to rank each display in order of helpfulness to complete the task (both moving the

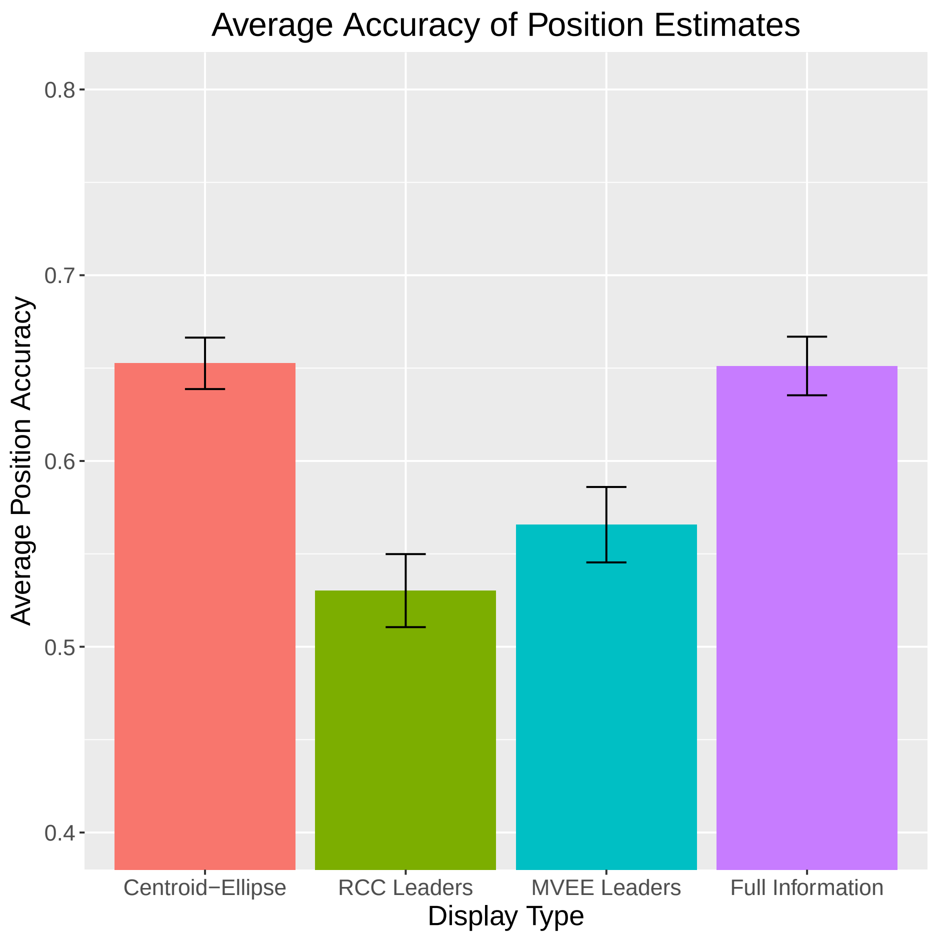

Figure 4.10: The average accuracy of the user-drawn position estimates for the swarm control study.

swarm to the goal and giving estimates). The subjective responses from the participants largely match the previous study, in that the Full Information and Centroid/Ellipse displays were both clearly superior to the MVEE and RCC Leader display modes. As before, the Full Information condition is slightly preferred over the Centroid/Ellipse, and MVEE Leaders seems more preferred over RCC. Interestingly, however, participants preferences did not correlate with their performance on the task, or their position and heading estimates.

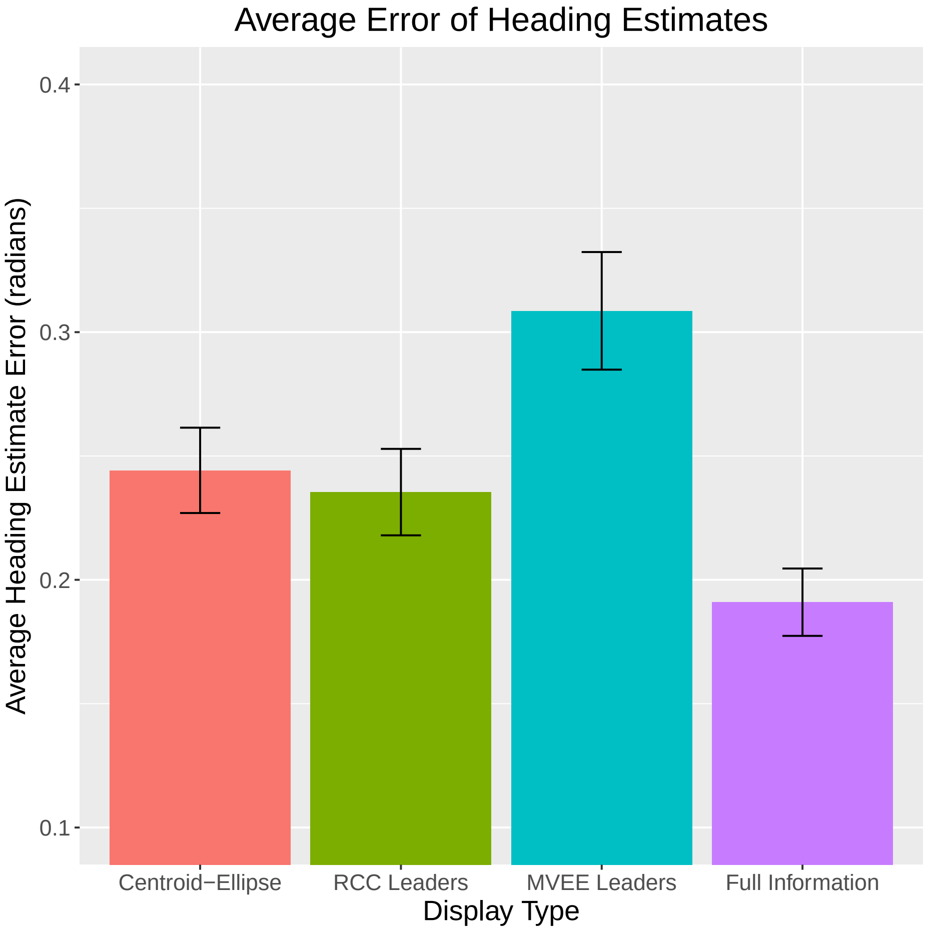

Figure 4.11: The average error of the user-drawn heading estimates during the swarm control study.

4.3.5 Discussion

Results from this study largely confirm, on average, those of the previous—that the Full Information and Centroid/Ellipse displays are preferred over leader-based displays for spatially oriented tasks in general. However, in this study the RCC Leader display actually performed significantly better than the MVEE display. Indeed, it was equally as effective as the Full Information and Centroid/Ellipse displays in the time taken to reach the goal, the average error of heading estimates, and in the average path length to reach the goal. Interestingly, the RCC Leader display was *not* as effective as Full Information or Centroid/Ellipse at the

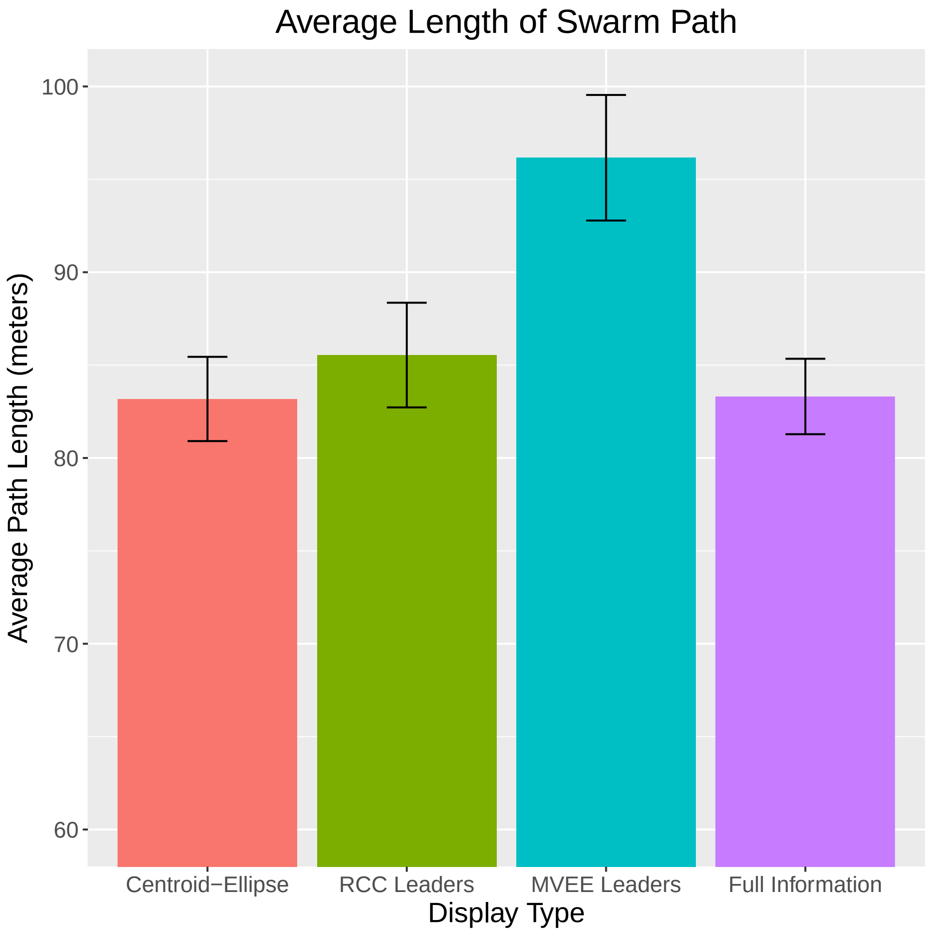

Figure 4.12: The average path length of swarm across a trial.

position estimates. Overall, this allows us to confirm **H1** and reject **H2**.

One possible explanation for the RCC Leader display not matching performance with Full Information and Centroid/Ellipse on position estimates is similar to the explanation given in the previous study for why dispersion gave the highest accuracy: for RCC Leaders, the bounds of the swarm are not easily recognizable, whereas for Full Information, Centroid/Ellipse, and MVEE Leaders the bounds are shown on the screen (either as robots in the first and third of these, and as a drawn ellipse in the second). What this means is that tasks in which the bounds of the swarm need to be known concretely should focus on a summarizing display, such as the Centroid/Ellipse methods, but that other tasks may be

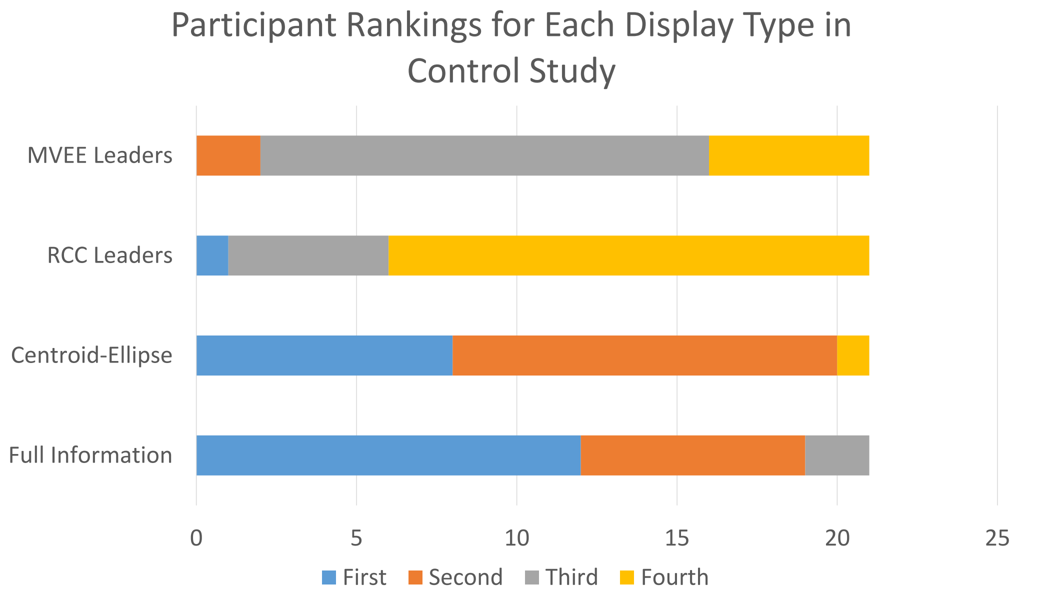

Figure 4.13: Participant rankings of each display type in the control study.

able to use RCC Leaders, or another leader selection method that keeps a relatively even distribution of leaders throughout the swarm.

The interesting finding for this study is that, in contrast to the prior study, RCC Leaders did quite well. One explanation for this is that because the task involved moving a swarm into a goal region, the strict bounds of the swarm were not as important, but rather the location and shape of the swarm. Because RCC Leaders selects robots evenly throughout the swarm, this information is still accessible to the user, although not as directly. MVEE Leaders, on the other hand, do provide some measure of spatial bounds of the swarm, but lacks information about the internal density of robots and in general makes using Gestalt properties to recognize swarm behavior difficult. Also interesting is that the participant rankings do not match their performance for the display. Participants ranked the RCC Leader method as least preferred, yet MVEE Leaders performed the worst. This suggests that users may feel less confident about or trusting toward the swarm under the RCC display method. While not necessarily detrimental to performance, this is still worrisome from the designer's perspective. Ideally, performance should match the user's trust or confidence in the swarm.

The final study, presented next, will begin to investigate this question of trust. Because the visualization of a swarm is the easiest way to impact user trust outside of modifying the swarm's behavior, particular care needs to be taken to ensure the display does not cause confusion or undesirably decrease the operator's trust.

4.4 FOURTH STUDY: SWARM TRUST

Visualizations of swarms can serve many purposes. The main one, as explored in the previous studies, is to improve operator understanding, prediction, and control of the emergent behaviors the swarm performs. A secondary purpose is to ensure the user has proper trust in the swarm. While performance and trust do not always correlate, undertrust or overtrust—where operators rely too much or too little on the automated system—is generally undesirable. Transparency, however, has been shown to increase trust in automation, and therefore it important to see if the visualizations of swarms—in effect, the vehicle for transparency—has the same effect as transparency in other types of automation.

The overall design of the study is similar to the previous task, with the main change being that user trust is now queried in pre- and post-questionnaires, as well as during the trials themselves. Furthermore, the task has changed from moving the swarm to a predefined goal region, to a more open-ended target searching task, which forces the user to interact with the swarm more, and provides greater opportunities for success or failure. The final difference is that rendezvous is used in conjunction with flocking—the main task will have users employing flocking most of the time, however rendezvous is a helpful tool to recollect the swarm if it becomes disconnected into many parts, which is undesirable for this particular task.

The parameters used for the flocking in this particular study are given in Table 4.5. The rendezvous algorithm used is different from the one presented in Section 3.1, however, and will be introduced in the following section.

Table 4.5: Simulation parameter values for trust study.

Variable	**Value**
d_1	2.5 meters
d_2	3.0 meters
r	6.0 meters
v_{max}	3.0 meters/sec
α_{max}	6π radians/sec
w_a	1.0 or 0.6
w_c	1.0
w_r	1.25

4.4.1 Study Design

The swarm consisted of 32 robots, which began at random positions and orientations within a 10×10 meter box centered at the origin of the simulation environment. The environment extended from ±200 meters in both the x- and y-dimension. The participants had two possible commands during a trial: change the goal heading of the leaders (done by dragging a line on the screen in the desired direction) during flocking, or switch to rendezvous, which sets the goal vector of each robot to the center of mass of the entire swarm. This allowed the user to reconnect disconnected parts of the swarm: by invoking the rendezvous command, the interface queries each robot for its position, and then computes the center point coordinates and returns that to the swarm. While this is not "swarmish" in the sense that it breaks the scalable, distributed nature of swarms, it is meant to be a rarely-used, emergency reconnect command. The environment always had 20 randomly-generated rectangular obstacles, each with a random length and width between 5 and 25 meters. Overlaps creating more complex, non-rectangular obstacles were allowed, but no obstacles were allowed within the swarm's starting space.

The main task involves moving the swarm throughout the environment to discover initially hidden targets. There are 100 targets total randomly dispersed throughout the free space of the environment. At each simulation time step, if a robot is within range of a target, one “unit” of information from that target is collected and relayed to the nearest leader to return to the display. Once 1,200 units of information about a target are gathered, the target is considered found and added to the total count discovered. Note that this is roughly equivalent to a single robot viewing a target for 20 seconds, or five robots viewing a target simultaneously for 4 seconds. This was used to mimic real-world scenarios where lots of information about a particular object or region need to be gathered. For instance, in human search and rescue scenarios expending resources to extract a person is costly, so an operator needs to be extra careful to ensure something reported by a robot is actually a victim in need of rescue, and not a piece of furniture or rubble that merely looks like a person. This also highlights the importance of keeping the swarm together: many robots together can identify targets quickly while still moving in a group. If instead the user spreads each robot out to its own exclusive region, more suspected targets may be seen initially, but it will take longer for each one to be identified. Furthermore, it will be more difficult to move a disconnected swarm to a new region, as they will either have to be reconnected or each commanded individually (which, as is the case in this study, is often not even possible), both of which are costly actions that consume a lot of the operator’s time.

The participants were separated into two groups, for a between-subjects design. The control group was shown a naïve display, where the robot positions, environment, and number of targets found were shown. The second group was called the “high transparency” group. This group received all the information of the control group, but in addition were shown the connections between the robots (i.e. a line is drawn between two robots on the display if they can communicate), as well as a graph on the side which shows the last one minute of statistics about the swarm size and number of targets found. Therefore, these participants could see when the swarm becomes larger (covers more area, but is at a higher risk of separation) along with when targets were found, and determine for themselves if there is a benefit to having a larger or smaller swarm. The display of the robots is shown in Figure 4.14 and the high transparency information graph in Figure 4.15.

Prior to the main experiment, participants filled out a survey asking about their general

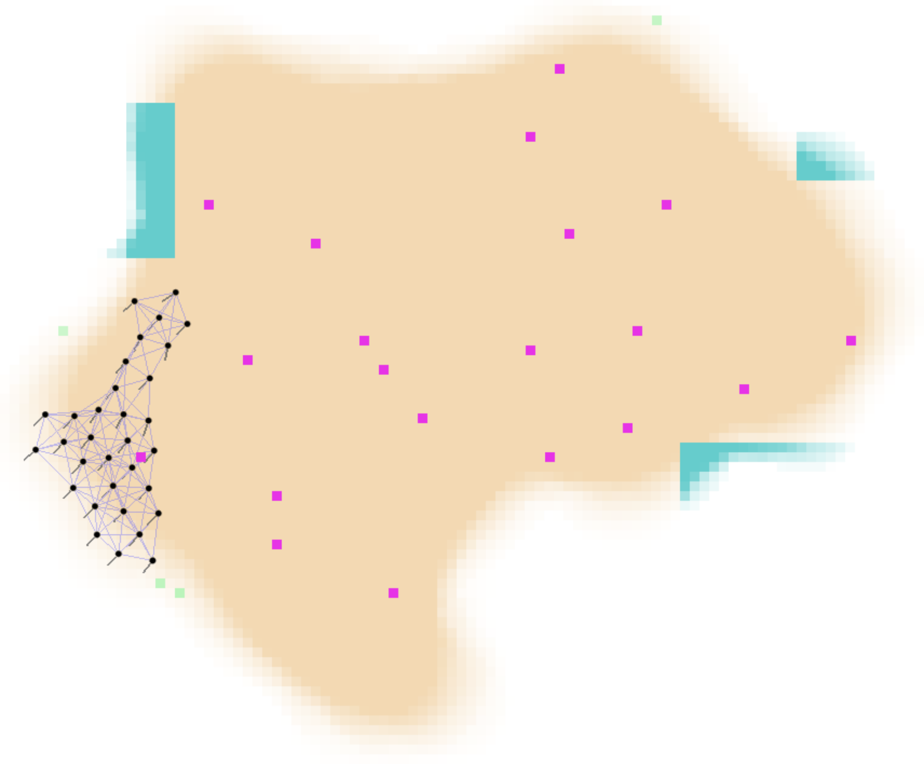

Figure 4.14: Display of the swarm and environment during the trust study. The robots are shown on the left in black, with gray lines between robots that can communicate with each other. The yellow region shows where the swarm has already explored. Identified, but unverified targets are seen at the edges of the explored region, and are shown in green. Verified targets are shown in purple. Obstacles are shown in teal. In the control group (low transparency) the robot connections are not shown.

trust in automation, adapted with permission from [18] (see Appendix), after which they had a chance to train on the simulator with the swarm by performing the main task for five minutes in their condition group (control or high transparency). The participants then completed three five-minute trials of the task described above in the assigned condition. The first trial is exactly as described above. The second of these trials was distinct in that, at five instances throughout the trial, the weight of the alignment vectors of every robot dropped

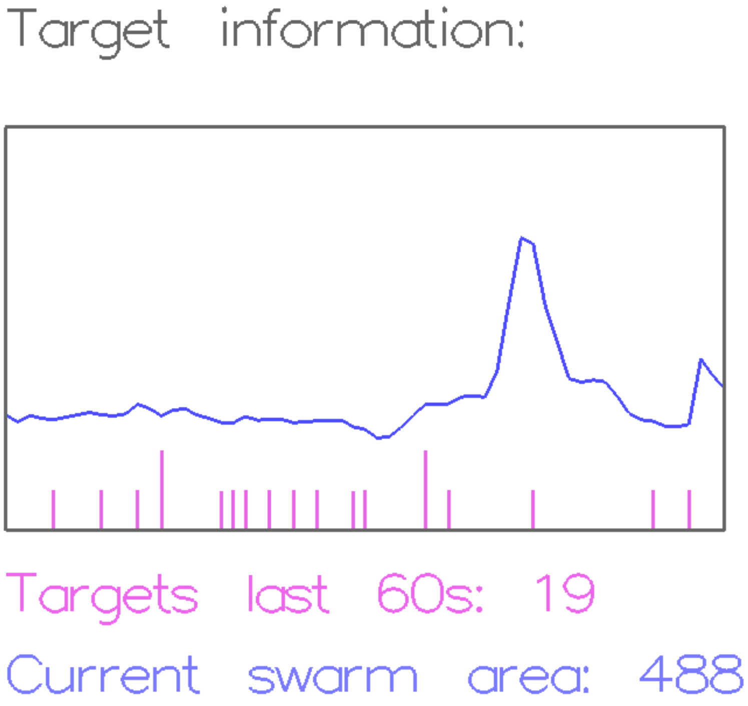

Figure 4.15: Information graph shown to participants in the high transparency conditions of the trust study. The blue line of the graph tracks the swarm size in m^2 (here currently $488m^2$), and the purple markers denote when targets were found (19 total in the previous 60 seconds). The graph tracks the last 60 seconds of the trial. In the control group (low transparency) the graph is not shown—only a text label displaying the total targets found so far is visible to the participant.

from 1.0 to 0.6. This was employed to mimic failures in robot sensing or movement that can often occur in real-world systems, to determine if trust is adversely affected. The failure points were randomly assigned so that one occured during each of the five minutes of the trial, no two failure points were within 30 seconds of each other and no failure point occured in the first 20 seconds of a trial. Having trials both with and without these failures allowed for a comparison of how high transparency impacts trust both when the swarm is performing correctly and when it is unreliable. In both of the first two conditions, the user was also queried for their trust value in the swarm at thirty-second intervals, on a scale from -10 (lowest trust value) to +10 (highest trust value). Participants were free to adjust this value, which is always present, at any time, but the mandatory update points were added to ensure the current trust value in the system—and shown on the display—accurately reflected the users current trust. The third and final trial were be identical to the second, except that participants were queried for trust after each command given, instead of every 30 seconds.

The main performance measure is the number of targets found during each of the five-minute trials. Secondary to that is the total area explored, which should correlate with targets found. Other secondary measures include the number of rendezvous and flocking commands given, as well as the total time spent performing each behavior. More time flocking will generally mean more targets found, as rendezvous typically does not move the swarm throughout new area, and thus is unlikely to lead to the discovery of new targets. Furthermore, these measures, in addition to the trust explicitly reported by the user, are useful for future work to build a model of user trust whereby the swarm itself can estimate the operator's current trust level and adapt its behavior accordingly. Such work is out of the scope of this thesis, but is useful to consider for future research nonetheless. The analysis will use these measures of performance to determine if the high transparency condition leads to better performance at the task or higher trust in the swarm. Furthermore, the data can be used to see if features of the swarm, such as heading variance of its members or total size taken up, correlate with higher or lower trust and performance.

After each of the trials, participants were given a specific trust questionnaire, similar to the general trust questionnaire and also adapted with permission from [18] (see Appendix). The specific trust questionnaire is divided into three categories of three questions each: Performance Expectancy, Process Transparency, and Purpose Influence. The participants also complete a NASA-TLX survey after each trial [47].

4.4.2 Participant Details

Participants were recruited from the University of Pittsburgh and surrounding area. There were 22 participants total, with an age range of 19-37 years old (median or 25). Of the participants, 11 were male and 11 were female.

4.4.3 Hypotheses

The primary two hypotheses of this study is that the high transparency condition will give both better performance and higher trust overall. This is due to the fact that the high transparency condition shows more information about the swarm's size and degree of separation, which should allow the users to feel more confident in their assessments of whether the swarm is going to stay together or is about to break apart. This higher degree of performance will mean more targets found and more area explored. Therefore, the hypotheses for this study are as follows:

- **H1**: The high transparency condition will give better performance (more targets found) on average.
- **H2**: The high transparency condition will lead to higher trust as reported by participants.

4.4.4 Results

The analysis first focuses on comparing the main performance measures between transparency conditions. The results found that those in the low transparency (control) group did no worse at finding targets than those in the high transparency group ($F(1, 51) = 2.84$, $p = .104$). If we relax the requirement to include targets seen but not fully explored, we find that those in the high transparency group found more targets, but only at a marginal significance level ($F(1, 51) = 3.65$, $p = .062$). The results also show that failures did not impact participants' abilities to find more targets, as there was no significant difference between trials with failures and those without ($F(1, 51) = 2.33$, $p = .133$). If we focus on explored area instead of targets found, we discover that those in the high transparency condition cover more area than those in the control condition, but again the results are only marginally significant ($F(1, 51) = 2.86$, $p = .097$).

If we look specifically at the results for transparency between failure and non-failure trials, we find nearly identical results. In the trials without failures, those in the high transparency group found statistically the same number of targets as those in the control group ($F(1,51) = 1.44$, $p = .241$). Relaxing the requirement to include targets seen but not fully explored, we similarly find that those in the high transparency condition perform better, but only at the marginally significant level ($F(1,51) = 3.55$, $p = .071$).

Moving on to the comparison of trust as reported by the participants, we find that there were no significant results between transparency and control conditions for the trust level reported by users during the trial ($F(1,51) = 2.46$, $p = .123$). Participants also report no differences in trust during the trials between failure and non-failure trials ($F(1,51) = 1.09$, $p = .301$, see Figure 4.16). However, we do find significant results when looking at the post-survey responses of user trust. Surprisingly though, those in the control condition reported *higher* trust on the surveys than those in the high transparency condition ($F(1,51) = 8.91$, $p = .004$). This seems to be driven largely by the increase in trust for Performance Expectancy ($F(1,51) = 4.26$, $p = .044$) and Purpose Influence ($F(1,51) = 13.24$, $p < .001$). Interestingly, the section of the survey focused on transparency—Process Transparency—gave no significant result ($F(1,51) = 2.17$, $p = .147$). Another interesting finding is that the results from the NASA-TLX survey given at the end of each trial show that participants reported a higher workload in the high transparency trials than in the control trials ($F(1,51) = 12.93$, $p < .001$, see Figure 4.17). However, a simple mediation analysis shows that the relationship between condition (high transparency vs. control) and trust was not mediated by workload. Significance of this indirect effect was tested using nonparametric bootstrapping procedures using 1,000 samples. The bootstrapped effect was measured at 0.27, and the 95% confidence intervals ranged from [-0.07, 0.89], $p = .180$.

Investigating trends for participants overall gives some significant correlations. Generally speaking, more targets found resulted in a higher level of trust as reported during the trials, although the effect was small and results only marginally significant ($t = 1.86$, $r^2 = 0.045$, $p = .068$). Similar results were found for all targets seen, not just those fully explored ($t = 1.74$, $r^2 = 0.037$, $p = .089$). By investigating features of the swarm we see that swarm size correlates with trust as reported during trials, although the effect was small and significance marginal ($t = 1.86$, $r^2 = 0.04$, $p = .068$). Average variance in swarm heading, however, more strongly correlated with trust during the trials, with higher variance giving lower trust ($t = 3.60$, $r^2 = 0.187$, $p < .001$).

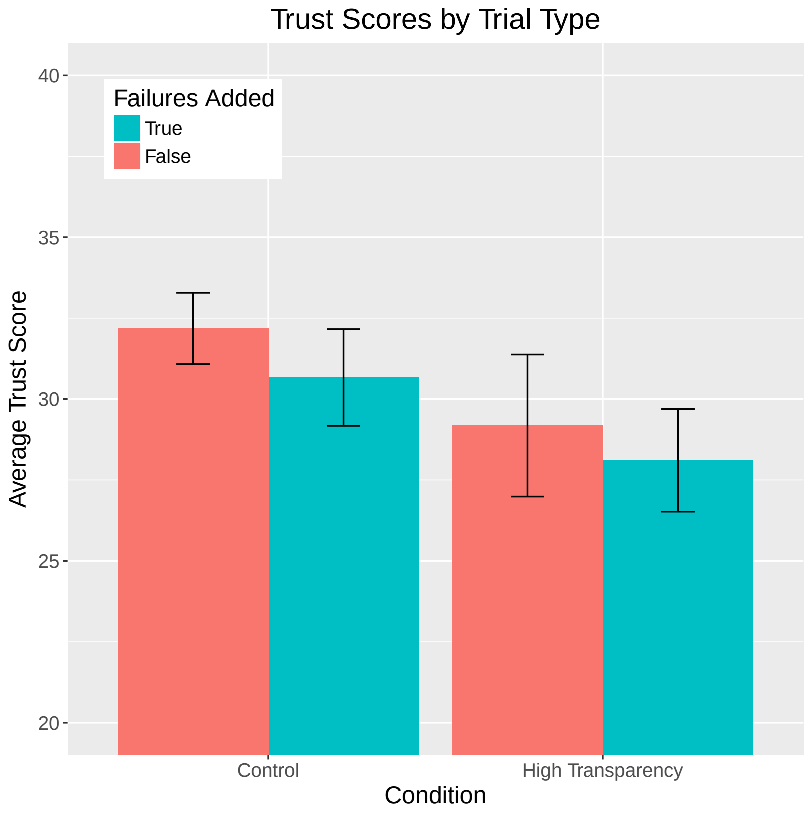

Figure 4.16: Average total trust scores given by users for the final trust survey.

Finally, if we look at the third trial, where participants reported their trust after each command issued, we can make comparisons between *trusted* and *untrusted* commands—that is, commands where the participants reported positive or neutral trust versus commands where participants reported negative trust (distrust). Here, we find that untrusted commands came during periods of higher variance in the swarm members' headings ($p < .001$) and larger physical swarm sizes ($p < .001$). No other swarm state features yielded significant results.

4.4.5 Discussion

The results clearly fail to confirm both hypotheses **H1** and **H2** in Section 4.4.3. Higher transparency failed to either improve performance or improve trust. In fact, the higher

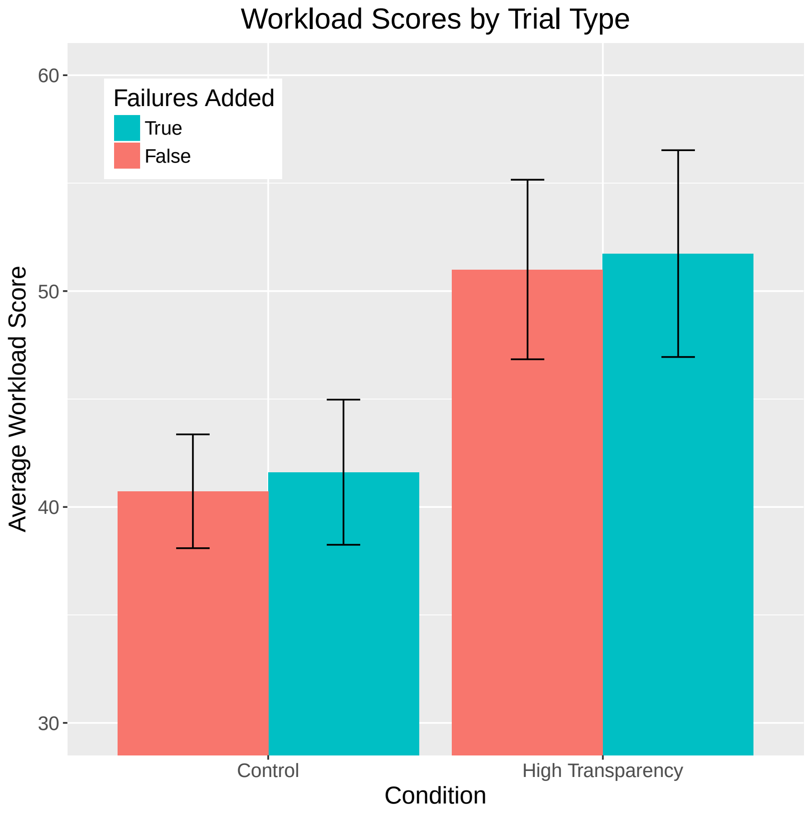

Figure 4.17: Average workload scores given by users for the NASA-TLX survey.

transparency actually led to lower trust in some cases. One possible explanation for the lack of improvement given by the higher transparency involves how the transparency was given to participants. In the experiment, extra information was shown on the left side of the screen, away from the main operation of the swarm. Responses given by participants after the experiments completed indicated many of them felt it was either unnecessary or distracting, as it drew their attention away from the operation of the swarm. Therefore, a conclusion that can be drawn is that if improved or more detailed visualization is to be added to a human-swarm interface, it should be overlaid or integrated with the swarm itself—having participants move their gaze between different disconnected areas of the interface seems to have a negative impact. This would explain particularly why trust dropped, as participants distracted by those extra visualization elements felt as though the system (which includes the interface) was deliberately making things more difficult.

The recommendation then is not that improved visualization is always a bad thing, but that it should be significantly limited and not distract from the main operation of the swarm. In future studies, designers of HSI interfaces should focus on displaying more information about the swarm within the swarm itself. For instance, to display variance in swarm headings, one could overlay a cone on the swarm whose width increases or decreases based on heading variance. Similarly, future studies could investigate this fact more deeply by asking participants about which elements the preferred and which they did not. It may be the case in the study herein that participants found displaying the connections between the robots as helpful, but the extra display on the side unhelpful.

5.0 CONCLUSIONS

The aim of this dissertation is to begin a line of researching investigating how displays of swarms for human operators can be improved, and what aspects of swarms need to be highlighted in such displays. Up until this point, there has been little to no research investigating these questions, and it is my hope that the work presented herein will help guide future research in the field.

Overall, the main conclusion is that the current behavior of a swarm has a significant impact on how operators view and respond to changes in the swarm's configuration. Some behaviors are easier to recognize and predict than others, and therefore different behaviors require some elements of the swarm state to be highlighted by a display, and other behaviors may require different elements. For instance, from the results presented herein, flocking (and likely other consensus-based behaviors) could benefit from summary information about the degree to which consensus has been reached. Furthermore, all behaviors seem to benefit from displays that keep the holistic, Gestalt nature of swarms intact. This research, along with the large body of work in human perception, give credence to the idea that large collectives of things moving in sync with each other tend to be viewed as a single group or single entity. Therefore, breaking this will likely be to the detriment of the main task at hand.

We can now begin to answer the questions posed at the beginning of this work:

1. *What variables or characteristics of a swarm are the most valuable to improving an operator's ability to understand and predict the swarm state?*

   The most important variables to show, from the results presented herein, are global variables describing the overall nature of the swarm. These are variables such as total swarm size or heading variance. Particularly, variables upon which the swarm is performing consensus may be of particular interest to operators of the swarm. Furthermore, displays of these variables should show them in a way that does not either detract from the singular, holistic nature of swarm movement or distract the user from the swarm and the task at hand.

2. *Can a visualization that maximizes operators' abilities to understand and predict the swarm also improve their ability to give proper control the swarm?*

   The results of the first three studies were remarkably similar, so it would seem that displays good at achieving proper recognition and prediction for users of the swarm's state are also useful for control tasks—at least with simple flocking-style control algorithms for completing the task. Future work should investigate whether this holds for other more complex tasks.

3. *Finally, can a visualization that achieves good results in accordance with the prior two questions also improve the operator's trust and confidence in the swarm, and is there a risk of a good visualization making operators too trusting?*

   Using the results of the first three studies, the fourth study aimed to focus on flocking, and display extra information about variables that seemed important from the prior studies: particularly that of swarm size and whether or not the swarm was cohesively composed as a single entity. However, results from the final study showed that these elements as they were displayed actually gave poorer performance. This does not mean that visualizations highlighting these seemingly important variables are a bad idea, but rather that special care should be taken to ensure they are not distracting from the main task. Future work should focus on limiting the cognitive impact of these extra interface elements and determining if smarter visualizations can indeed improve performance.

There are numerous avenues for future work to build off of the results presented here. Primarily, researchers in human-swarm interaction should focus on integrating elements of the display with the swarm itself, and investigate how to best overlay important global data that may not be immediately evident by viewing the swarm by itself. Furthermore, research should expand this work to include other types of tasks and behaviors to see if the results here translate more generally to HSI, or are specific to certain types of tasks. Finally, and most importantly, as the cost of robots and sensor continue to decrease, researchers should move this work to robots performing desired tasks in the real world. Moving from simulation to actual implementation always presents unforeseen problems, and answering questions about how to best display information about a swarm to its operator requires real-world swarms to fully answer. The important takeaway from this work, however, is that visualization of swarms is considerably more complex than single robots or multiple independently-operating robots, and that smarter visualization of swarms does not always mean more visualization. The important thing that any designer of an HSI interface should remember is not to break the natural appeal and easy recognition of coordinating collectives of agents. Humans have been viewing such collectives in flocks of birds and schools of fish for millennia, and we should use this to our advantage when displaying artificial swarms.

APPENDIX

ITEMS IN TRUST QUESTIONNAIRES

The following tables show the trust questionnaires given in the final experiment. These questionnaires are adapted from those first presented in [18].

Table A1: General trust questionnaire - given *before* experiment.

Performance Expectancy
Allowing a computer to automate tasks increases my effectiveness
Allowing a computer to automate tasks improves my output quality
Allowing a computer to automate tasks increases my chances of achieving higher performance
Process Transparency
The information provided by computers when automating tasks is generally of high quality
Computers provide sufficient information about their automation process
I am satisfied with the information provided by computers when they automate tasks
Cultural-Technological Context
I prefer to use a computer to automate tasks under high workload situations
Using a computer to automate tasks allows me to expend less effort
Using a computer to automate tasks poses less risk than performing the task manually

Table A2: Specific trust questionnaire - given *after* experiment.

Performance Expectancy
The automated control laws improved my performance
The control laws enabled me to accomplish the task more quickly
The control laws increased my productivity
Process Transparency
My interaction with the swarm was understandable
Interacting with the control laws was user-friendly
The control laws made the correct choices while the swarm was moving
Purpose Influence
I am confident about the performance of the control laws
When a difficult issue arose, I felt comfortable depending on the control laws to guide the swarm
I could rely on the control laws to ensure good performance

BIBLIOGRAPHY

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.

[2] E. Alberdi, A. Povyakalo, L. Strigini, and P. Ayton. Effects of incorrect computer-aided detection (cad) output on human decision-making in mammography. *Academic radiology*, 11(8):909–918, 2004.

[3] R. Arkin. *Behavior-based robotics*. The MIT Press, 1998.

[4] J. C. Barca and Y. A. Sekercioglu. Swarm robotics reviewed. *Robotica*, 31(03):345–359, 2013.

[5] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic. *Mobile ad hoc networking*. John Wiley & Sons, 2004.

[6] S. Bashyal and G. Venayagamoorthy. Human swarm interaction for radiation source search and localization. In *Swarm Intelligence Symposium, 2008. SIS 2008. IEEE*, pages 1–8. IEEE, 2008.

[7] L. Bayindir and E. Sahin. A review of studies in swarm robotics. *Turk. J. Elec. Engin*, 15(2):115–147, 2007.

[8] E. Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7280–7287, 2002.

[9] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm intelligence: from natural to artificial systems*, volume 4. Oxford university press New York, 1999.

[10] M. W. Boyce, J. Y. Chen, A. R. Selkowitz, and S. G. Lakhmani. Effects of agent transparency on operator trust. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, HRI'15 Extended Abstracts, pages 179–180, New York, NY, USA, 2015. ACM.

[11] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[12] D. S. Brown and M. A. Goodrich. Limited bandwidth recognition of collective behaviors in bio-inspired swarms. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 405–412. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[13] D. S. Brown, S. C. Kerman, and M. A. Goodrich. Human-swarm interactions based on managing attractors. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pages 90–97. ACM, 2014.

[14] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009. To appear. Electronically available at http://coordinationbook.info.

[15] S. Camazine. *Self-organization in biological systems*. Princeton University Press, 2003.

[16] H. Çelikkanat and E. Şahin. Steering self-organized robot flocks through externally guided individuals. *Neural Computing and Applications*, 19(6):849–865, 2010.

[17] Y. Cheung and J. Chung. Cooperative control of a multi-arm system using semi-autonomous telemanipulation and adaptive impedance. In *Advanced Robotics, 2009. ICAR 2009. International Conference on*, pages 1–7. IEEE, 2009.

[18] S.-Y. Chien, Z. Semnani-Azad, M. Lewis, and K. Sycara. Towards the development of an inter-cultural scale to measure trust in automation. In *International Conference on Cross-Cultural Design*, pages 35–46. Springer, 2014.

[19] J. Clark and R. Fierro. Cooperative hybrid control of robotic sensors for perimeter detection and tracking. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 3500–3505. IEEE, 2005.

[20] J. Cortés, S. Martínez, and F. Bullo. Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *Automatic Control, IEEE Transactions on*, 51(8):1289–1298, 2006.

[21] A. Couture-Beil, R. Vaughan, and G. Mori. Selecting and commanding individual robots in a multi-robot system. In *Computer and Robot Vision (CRV), 2010 Canadian Conference on*, pages 159 –166, 31 2010-june 2 2010.

[22] I. Couzin, J. Krause, R. James, G. Ruxton, and N. Franks. Collective memory and spatial sorting in animal groups. *Journal of theoretical biology*, 218(1):1–11, 2002.

[23] I. D. Couzin and J. Krause. Self-organization and collective behavior in vertebrates. *Advances in the Study of Behavior*, 32:1–75, 2003.

[24] M. Daily, Y. Cho, K. Martin, and D. Payton. World embedded interfaces for human-robot interaction. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*. IEEE, 2003.

[25] M. De Jongh, P. M. Dudas, and P. Brusilovsky. Adaptive visualization of research communities. In *Proceedings of the 1st International Workshop on User-Adaptive Visualization*. University of Pittsburgh, 2013.

[26] W. H. Dittrich, T. Troscianko, S. E. Lea, and D. Morgan. Perception of emotion from dynamic point-light displays represented in dance. *Perception-London*, 25(6):727–738, 1996.

[27] M. Dorigo, D. Floreano, L. M. Gambardella, F. Mondada, S. Nolfi, T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, et al. Swarmanoid. *IEEE Robotics & Automation Magazine*, 1070(9932/13), 2013.

[28] M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, et al. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2-3):223–245, 2004.

[29] P. M. Dudas, M. de Jongh, and P. Brusilovsky. A semi-supervised approach to visualizing and manipulating overlapping communities. In *Information Visualisation (IV), 2013 17th International Conference*, pages 180–185. IEEE, 2013.

[30] M. T. Dzindolet, L. G. Pierce, H. P. Beck, and L. A. Dawe. The perceived utility of human and automated aids in a visual detection task. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(1):79–94, 2002.

[31] A. Farinelli, L. Iocchi, and D. Nardi. Multirobot systems: a classification focused on coordination. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 34(5):2015 –2028, Oct. 2004.

[32] J. D. Farmer and D. Foley. The economy needs agent-based modelling. *Nature*, 460(7256):685–686, 2009.

[33] E. Ferrante, A. E. Turgut, C. Huepe, A. Stranieri, C. Pinciroli, and M. Dorigo. Self-organized flocking with a mobile robot swarm: a novel motion control method. *Adaptive Behavior*, 20(6):460–477, 2012.

[34] R. Fox and C. McDaniel. The perception of biological motion by human infants. *Science*, 218(4571):486–487, 1982.

[35] A. Franchi, P. R. Giordano, C. Secchi, H. I. Son, and H. H. Bulthoff. A passivity-based decentralized approach for the bilateral teleoperation of a group of uavs with switching topology. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 898–905. IEEE, 2011.

[36] A. Franchi, C. Masone, H. Bulthoff, and P. Robuffo Giordano. Bilateral teleoperation of multiple uavs with decentralized bearing-only formation control. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2215–2222. IEEE, 2011.

[37] A. Ganguli, J. Cortés, and F. Bullo. Visibility-based multi-agent deployment in orthogonal environments. In *American Control Conference, 2007. ACC'07*, pages 3426–3431. IEEE, 2007.

[38] P. Giordano, A. Franchi, C. Secchi, and H. Bülthoff. Bilateral teleoperation of groups of uavs with decentralized connectivity maintenance. In H. F. Durrant-Whyte, N. Roy, and P. Abbeel, editors, *Robotics: Science and Systems*, 2011.

[39] A. Giusti, J. Nagi, L. M. Gambardella, S. Bonardi, and G. A. Di Caro. Human-swarm interaction through distributed cooperative gesture recognition. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 401–402. ACM, 2012.

[40] M. Goodrich, B. Pendleton, P. Sujit, and J. Pinto. Toward human interaction with bio-inspired robot teams. In *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, pages 2859–2864. IEEE, 2011.

[41] M. Goodrich, P. Sujit, S. Kerman, B. Pendleton, and J. Pinto. Enabling human interaction with bio-inspired robot teams: Topologies, leaders, predators, and stakeholders byu-hcmi technical report 2011-1. Technical report, Brigham Young University, 2011.

[42] M. A. Goodrich, S. Kerman, and S.-Y. Jung. On leadership and influence in human-swarm interaction. In *2012 AAAI Fall Symposium Series*, 2012.

[43] M. A. Goodrich, B. Pendleton, S. Kerman, and P. Sujit. What types of interactions do bio-inspired robot swarms and flocks afford a human? In *Robotics: Science and Systems VIII*, pages 105–112. MIT Press, 2013.

[44] V. Grimm, E. Revilla, U. Berger, F. Jeltsch, W. M. Mooij, S. F. Railsback, H.-H. Thulke, J. Weiner, T. Wiegand, and D. L. DeAngelis. Pattern-oriented modeling of agent-based complex systems: lessons from ecology. *science*, 310(5750):987–991, 2005.

[45] E. Grossman, M. Donnelly, R. Price, D. Pickens, V. Morgan, G. Neighbor, and R. Blake. Brain areas involved in perception of biological motion. *Journal of cognitive neuroscience*, 12(5):711–720, 2000.

[46] E. Haas, M. Fields, S. Hill, and C. Stachowiak. Extreme scalability: Designing interfaces and algorithms for soldier-robotic swarm interaction. Technical report, DTIC Document, 2009.

[47] S. Hart and L. Staveland. Development of a multi-dimensional workload rating scale: Results of empirical and theoretical research. in, pa hancock and n. meshkati. *Human Mental Workload*, 1988.

[48] S. T. Hayes and J. A. Adams. Human-swarm interaction: sources of uncertainty. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction*, pages 170–171. ACM, 2014.

[49] J. Heer and D. Boyd. Vizster: Visualizing online social networks. In *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on*, pages 32–39. IEEE, 2005.

[50] H. Hexmoor, S. Eluru, and H. Sabaa. Plan sharing: Showcasing coordinated uav formation flight. *INFORMATICA*, 30(2):183, 2006.

[51] H. Hexmoor, B. McLaughlan, and M. Baker. Swarm control in unmanned aerial vehicles. In *Proceedings of the International Conference on Artificial Intelligence*, pages 911–917, 2005.

[52] P. Holmes, R. Full, D. Koditschek, and J. Guckenheimer. The dynamics of legged locomotion: Models, analyses, and challenges. *Siam Review*, pages 207–304, 2006.

[53] A. Howard, M. Mataric, and G. Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In H. Asama, T. Arai, T. Fukuda, and T. Hasegawa, editors, *Distributed Autonomous Robotic Systems 5*, volume 5, pages 299–308. http://dx.doi.org/10.1007/978–4–431–65941–9_30, Fukuoka, Japan, 2002. Springer Japan.

[54] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *Automatic Control, IEEE Transactions on*, 48(6):988–1001, 2003.

[55] G. Johansson. Visual perception of biological motion and a model for its analysis. *Attention, Perception, & Psychophysics*, 14(2):201–211, 1973.

[56] K. N. Kaipa, A. Puttappa, G. M. Hegde, S. V. Bidargaddi, and D. Ghose. Rendezvous of glowworm-inspired robot swarms at multiple source locations: A sound source based real-robot implementation. In *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pages 259–269. Springer, 2006.

[57] I. Kassabalidis, M. El-Sharkawi, R. Marks, P. Arabshahi, A. Gray, et al. Swarm intelligence for routing in communication networks. In *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, volume 6, pages 3613–3617. IEEE, 2001.

[58] S. Kerman, D. Brown, and M. Goodrich. Supporting human interaction with robust robot swarms. In *Resilient Control Systems (ISRCS), 2012 5th International Symposium on*, pages 197–202. IEEE, 2012.

[59] Z. Kira and M. Potter. Exerting human control over decentralized robot swarms. In *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on*, pages 566–571. IEEE, 2009.

[60] A. Kolling, S. Nunnally, and M. Lewis. Towards human control of robot swarms. In *Proceedings of the 7th international conference on Human-robot interaction*, pages 89–96. ACM, 2012.

[61] A. Kolling, K. Sycara, S. Nunnally, and M. Lewis. Human swarm interaction: An experimental study of two types of interaction with foraging swarms. *Journal of Human-Robot Interaction*, 2(2):103–128, 2013.

[62] D. Kornhauser, U. Wilensky, and W. Rand. Design guidelines for agent based model visualization. *Journal of Artificial Societies and Social Simulation*, 12(2):1, 2009.

[63] K. Krishnanand and D. Ghose. Theoretical foundations for rendezvous of glowworm-inspired agent swarms at multiple locations. *Robotics and Autonomous Systems*, 56(7):549–569, 2008.

[64] J. LaSalle. *The Stability of Dynamical Systems*, volume 25. SIAM, 1976.

[65] C. Layton, P. J. Smith, and C. E. McCoy. Design of a cooperative problem-solving system for en-route flight planning: An empirical evaluation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 36(1):94–119, 1994.

[66] D. Lee, A. Franchi, P. Giordano, H. Son, and H. Bulthoff. Haptic teleoperation of multiple unmanned aerial vehicles over the internet. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1341–1347. IEEE, 2011.

[67] J. D. Lee and K. A. See. Trust in automation: Designing for appropriate reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 46(1):50–80, 2004.

[68] T. Lenox, M. Lewis, E. Roth, R. Shern, L. Roberts, T. Rafalski, and J. Jacobson. Support of teamwork in human-agent teams. In *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, volume 2, pages 1341–1346. IEEE, 1998.

[69] N. E. Leonard, D. A. Paley, R. E. Davis, D. M. Fratantoni, F. Lekien, and F. Zhang. Coordinated control of an underwater glider fleet in an adaptive ocean sampling field experiment in monterey bay. *Journal of Field Robotics*, 27(6):718–740, 2010.

[70] M. Lewis. Human interaction with multiple remote robots. *Reviews of Human Factors and Ergonomics*, 9(1):131–174, 2013.

[71] W. Luo, S. S. Khatib, S. Nagavalli, N. Chakraborty, and K. Sycara. Asynchronous distributed information leader selection in robotic swarms. In *IEEE International Conference on Automation Science and Engineering*, August 2015.

[72] J. B. Lyons and C. K. Stokes. Human–human reliance in the context of automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, page 0018720811427034, 2011.

[73] P. K. Maini, H. G. Othmer, et al. *Mathematical models for biological pattern formation.* Springer, 2001.

[74] G. Mather and L. Murdoch. Gender discrimination in biological motion displays based on dynamic cues. *Proceedings of the Royal Society of London B: Biological Sciences*, 258(1353):273–279, 1994.

[75] S. Mau and J. Dolan. Scheduling for humans in multirobot supervisory control. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1637–1643. IEEE, 2007.

[76] R. C. Mayer, J. H. Davis, and F. D. Schoorman. An integrative model of organizational trust. *Academy of management review*, 20(3):709–734, 1995.

[77] J. M. McGuirl and N. B. Sarter. Supporting trust calibration and the effective use of decision aids by presenting dynamic system confidence information. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(4):656–665, 2006.

[78] K. A. McKibbon and D. B. Fridsma. Effectiveness of clinician-selected electronic information resources for answering primary care physicians information needs. *Journal of the American Medical Informatics Association*, 13(6):653–659, 2006.

[79] J. McLurkin and J. Smith. Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots. In *Distributed Autonomous Robotic Systems 6*, pages 399–408. Springer Japan, 2007.

[80] C. A. Miller and R. Parasuraman. Designing for flexible interaction between humans and automation: Delegation interfaces for supervisory control. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49(1):57–75, 2007.

[81] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. Swarm-bot: A new distributed robotic concept. *Autonomous robots*, 17(2-3):193–221, 2004.

[82] K. L. Mosier, E. A. Palmer, and A. Degani. Electronic checklists: Implications for decision making. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 36, pages 7–11. SAGE Publications, 1992.

[83] K. L. Mosier and L. J. Skitka. Human decision makers and automated decision aids: Made for each other? *Automation and human performance: Theory and applications*, pages 201–220, 1996.

[84] K. L. Mosier, L. J. Skitka, S. Heers, and M. Burdick. Automation bias: Decision making and performance in high-tech cockpits. *The International journal of aviation psychology*, 8(1):47–63, 1998.

[85] S. Nagavalli, N. Chakraborty, and K. Sycara. A study on the bounds of neglect benevolence in input timing for human interaction with robotic swarms. In *Human-Robot Interaction (HRI), 2015 ACM/IEEE International Conference on*, 2015.

[86] S. Nagavalli, S.-Y. Chien, M. Lewis, N. Chakraborty, and K. Sycara. Bounds of neglect benevolence in input timing for human interaction with robotic swarms. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, pages 197–204. ACM, 2015.

[87] S. Nagavalli, L. Luo, N. Chakraborty, and K. Sycara. Neglect benevolence in human control of robotic swarms. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6047–6053, May 2014.

[88] J. Nagi, A. Giusti, L. M. Gambardella, and G. A. Di Caro. Human-swarm interaction using spatial gestures. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3834–3841. IEEE, 2014.

[89] J. Nickolls, I. Buck, M. Garland, and K. Skadron. Scalable parallel programming with cuda. *Queue*, 6(2):40–53, 2008.

[90] S. Nunnally, P. Walker, A. Kolling, N. Chakraborty, M. Lewis, K. Sycara, and M. Goodrich. Human influence of robotic swarms with bandwidth and localization issues. *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pages 333–338, 2012.

[91] S. Nunnally, P. Walker, M. Lewis, N. Chakraborty, and K. Sycara. Using haptic feedback in human robotic swarms interaction. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 57, pages 1047–1051. SAGE Publications, 2013.

[92] R. O'Grady, R. Groß, A. L. Christensen, and M. Dorigo. Self-assembly strategies in a group of autonomous mobile robots. *Autonomous Robots*, 28(4):439–455, 2010.

[93] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *Automatic Control, IEEE Transactions on*, 51(3):401–420, 2006.

[94] D. Olsen and M. Goodrich. Metrics for evaluating human-robot interactions. In *Proceedings of PERMIS*, volume 2003, 2003.

[95] M. W. Oram, D. Perrett, et al. Responses of anterior superior temporal polysensory (stpa) neurons to biological motion stimuli. *Cognitive Neuroscience, Journal of*, 6(2):99–116, 1994.

[96] R. Parasuraman and V. Riley. Humans and automation: Use, misuse, disuse, abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 39(2):230–253, 1997.

[97] L. E. Parker. Multiple mobile robot systems. *Springer Handbook of Robotics*, pages 921–941, 2008.

[98] D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee. Pheromone robotics. *Autonomous Robots*, 11(3):319–324, 2001.

[99] B. Pendleton and M. Goodrich. Scalable human interaction with robotic swarms. In *"proceedings of the AIAA Infotech Aerospace Conference"*, 2013.

[100] C. E. Perkins. *Ad hoc networking*. Addison-Wesley Professional, 2008.

[101] R. S. Peterson and E. G. Sirer. Antfarm: Efficient content distribution with managed swarms. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'09, pages 107–122, Berkeley, CA, USA, 2009. USENIX Association.

[102] J. Pinto and M. Shiffrar. The visual perception of human and animal motion in point-light displays. *Social Neuroscience*, 4(4):332–346, 2009.

[103] S. Pourmehr, V. M. Monajjemi, R. Vaughan, and G. Mori. "you two! take off!": Creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 137–142. IEEE, 2013.

[104] C. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH Computer Graphics*, volume 21, pages 25–34. ACM, 1987.

[105] E. J. Rodríguez-Seda, J. J. Troy, C. A. Erignac, P. Murray, D. M. Stipanovic, and M. W. Spong. Bilateral teleoperation of multiple mobile agents: Coordinated motion and collision avoidance. *IEEE Transactions on Control Systems Technology*, 18(4):984–992, 2010.

[106] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal. Kilobot: A low cost robot with scalable operations designed for collective behaviors. *Robotics and Autonomous Systems*, 62(7):966–975, 2014.

[107] M. Rubenstein, A. Cornejo, and R. Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799, 2014.

[108] E. Şahin. Swarm robotics: From sources of inspiration to domains of application. In *Swarm robotics*, pages 10–20. Springer, 2005.

[109] M. Saleem, G. A. Di Caro, and M. Farooq. Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions. *Information Sciences*, 181(20):4597–4624, 2011.

[110] F. Salim, M. Williams, N. Sony, M. Dela Pena, Y. Petrov, A. A. Saad, and B. Wu. Visualization of wireless sensor networks using zigbee's received signal strength indicator (rssi) for indoor localization and tracking. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 575–580. IEEE, 2014.

[111] N. B. Sarter and B. Schroeder. Supporting decision making and action selection under time pressure and uncertainty: The case of in-flight icing. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 43(4):573–583, 2001.

[112] J. Sauter, S. Robert, A. Mathews, and S. Riddle. Distributed pheromone-based swarming control of unmanned air and ground vehicles for rsta. *Ann Arbor*, 1001:48105, 2008.

[113] J. A. Sauter, R. S. Matthews, J. S. Robinson, J. Moody, and S. P. Riddle. Swarming unmanned air and ground systems for surveillance and base protection. In *Proceedings of AIAA Infotech@ Aerospace 2009 Conference*, 2009.

[114] C. Secchi, A. Franchi, H. Bülthoff, and P. Giordano. Bilateral teleoperation of a group of uavs with communication delays and switching topology. *Proceedings of the IEEE International Conference on Robotics and Automation*, 2012.

[115] J. L. Segovia-Juarez, S. Ganguli, and D. Kirschner. Identifying control mechanisms of granuloma formation during m. tuberculosis infection using an agent-based model. *Journal of theoretical biology*, 231(3):357–376, 2004.

[116] A. Seiffert, S. Hayes, C. Harriott, and J. Adams. Motion perception of biological swarms. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society*, pages 2128–2133. Cognitive Science Society, 2015.

[117] A. Shklarsh, G. Ariel, E. Schneidman, and E. Ben-Jacob. Smart swarms of bacteria-inspired agents with performance adaptable interactions. *PLoS Comput Biol*, 7(9):e1002177, 2011.

[118] S. Sirouspour. Modeling and control of cooperative teleoperation systems. *Robotics, IEEE Transactions on*, 21(6):1220–1225, 2005.

[119] M. Steinberg. Biologically-inspired approaches for self-organization, adaptation, and collaboration of heterogeneous autonomous systems. In *Proceedings of SPIE*, volume 8062, page 80620H, 2011.

[120] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich. Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 33–40. ACM, 2006.

[121] A. Steinfeld, O. C. Jenkins, and B. Scassellati. The oz of wizard: simulating the human for interaction research. In *Human-Robot Interaction (HRI), 2009 4th ACM/IEEE International Conference on*, pages 101–107. IEEE, 2009.

[122] D. P. Stormont. Autonomous rescue robot swarms for first responders. In *CIHSPS 2005. Proceedings of the 2005 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety, 2005.*, pages 151–157. IEEE, 2005.

[123] F. Stürzel and L. Spillmann. Perceptual limits of common fate. *Vision research*, 44(13):1565–1573, 2004.

[124] D. Sumpter. The principles of collective animal behaviour. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 361(1465):5–22, 2006.

[125] K. Sycara, C. Lebiere, Y. Pei, D. Morrison, Y. Tang, and M. Lewis. Abstraction of analytical models from cognitive models of human control of robotic swarms. In *Proceedings of the Thirteenth International Conference on Cognitive Modeling, Groningen, Germany*, 2015.

[126] B. Tabibian, M. Lewis, C. Lebiere, N. Chakraborty, K. Sycara, S. Bennati, and M. Oishi. Towards a cognitively-based analytic model of human control of swarms. In *2014 AAAI Spring Symposium Series*, 2014.

[127] Y. Tan and Z.-Y. Zheng. Research advance in swarm robotics. *Defence Technology*, 9(1):18 – 39, 2013.

[128] H. Tanner, A. Jadbabaie, and G. Pappas. Stable flocking of mobile agents, part i: Fixed topology. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 2, pages 2010–2015. IEEE, 2003.

[129] C. Torney, Z. Neufeld, and I. D. Couzin. Context-dependent interaction leads to emergent search behavior in social aggregates. *Proceedings of the National Academy of Sciences*, 106(52):22055–22060, 2009.

[130] V. Trianni, R. Groß, T. H. Labella, E. Şahin, and M. Dorigo. Evolving aggregation behaviors in a swarm of robots. In *Advances in artificial life*, pages 865–874. Springer, 2003.

[131] W. Truszkowski, M. Hinchey, J. Rash, and C. Rouff. Nasa's swarm missions: The challenge of building autonomous software. *IT professional*, 6(5):47–52, 2004.

[132] W. R. Uttal, L. Spillmann, F. Stürzel, and A. B. Sekuler. Motion and shape in common fate. *Vision Research*, 40(3):301–310, 2000.

[133] R. Vaughan, N. Sumpter, J. Henderson, A. Frost, and S. Cameron. Experiments in automatic flock control. *Robotics and Autonomous Systems*, 31(1):109–117, 2000.

[134] P. Walker, S. Amirpour Amraii, N. Chakraborty, M. Lewis, and K. Sycara. Human control of robot swarms with dynamic leaders. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 1108–1113. IEEE, 2014.

[135] P. Walker, S. Amirpour Amraii, M. Lewis, N. Chakraborty, and K. Sycara. Control of swarms with multiple leader agents. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 3567–3572. IEEE, 2014.

[136] P. Walker, S. A. Amraii, M. Lewis, N. Chakraborty, and K. Sycara. Human control of leader-based swarms. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 2712–2717. IEEE, 2013.

[137] P. Walker, S. Nunnally, M. Lewis, N. Chakraborty, and K. Sycara. Levels of automation for human influence of robot swarms. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 57, pages 429–433. SAGE Publications, 2013.

[138] P. Walker, S. Nunnally, M. Lewis, A. Kolling, N. Chakraborty, and K. Sycara. Neglect benevolence in human control of swarms in the presence of latency. In *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pages 3009–3014. IEEE, 2012.

[139] B. Walter, A. Sannier, D. Reiners, and J. Oliver. Uav swarm control: Calculating digital pheromone fields with the gpu. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 3(3):167–176, 2006.

[140] N. Wang, D. V. Pynadath, and S. G. Hill. Trust calibration within a human-robot team: Comparing automatically generated explanations. In *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, HRI '16, pages 109–116, Piscataway, NJ, USA, 2016. IEEE Press.

[141] K. Xu and M. Gerla. A heterogeneous routing protocol based on a new stable clustering scheme. In *MILCOM 2002. Proceedings*, volume 2, pages 838–843. IEEE, 2002.

[142] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[143] J. Yu, S. M. LaValle, and D. Liberzon. Rendezvous without coordinates. *Automatic Control, IEEE Transactions on*, 57(2):421–434, 2012.