2013-03-08

Toward Scalable Human Interaction with Bio-Inspired Robot Teams

Daniel Sundquist Brown
Brigham Young University - Provo

Toward Scalable Human Interaction with Bio-Inspired Robot Teams

Daniel S. Brown

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Michael A. Goodrich, Chair
Jeffrey C. Humpherys
Tony R. Martinez

Department of Computer Science

Brigham Young University

March 2013

ABSTRACT

Toward Scalable Human Interaction with Bio-Inspired Robot Teams

Daniel S. Brown
Department of Computer Science, BYU
Master of Science

Bio-inspired swarming behaviors provide an effective decentralized way of coordinating robot teams. However, as robot swarms increase in size, bandwidth and time constraints limit the number of agents a human can communicate with and control. To facilitate scalable human interaction with large robot swarms it is desirable to monitor and influence the collective behavior of the entire swarm through limited interactions with a small subset of agents. However, it is also desirable to avoid situations where a small number of agent failures can adversely affect the collective behavior of the swarm. We present a robust bio-inspired model of swarming that exhibits distinct collective behaviors and affords limited human interaction to estimate and influence these collective behaviors. Using a simple naive Bayes classifier, we show that the global behavior of a swarm can be detected with high accuracy by sampling local information from a small number of agents. We also show that adding a bio-inspired form of quorum sensing to a swarm increases the scalability of human-swarm interactions and also provides an adjustable threshold on the swarm's vulnerability to agent failures.

Keywords: Human robot interaction, Bio-inspired swarms, Observation effort, Quorum sensing

ACKNOWLEDGMENTS

I'd like to thank my advisor, Dr. Mike Goodrich, for his guidance and suggestions, my committee members for their time and effort, and everyone in the BYU-HCMI lab for their support and encouragement. A special thanks is due to my wife Sarah for her love and support despite the many late nights and early mornings spent writing this thesis and to my parents and siblings for their encouragement and help.

Table of Contents

List of Figures **ix**

List of Tables **xvii**

1 Introduction **1**

2 Related Work **4**

2.1 Models of swarming behavior 4

2.2 Types of swarm behavior 5

2.3 Detection and estimation of collective behavior 7

2.4 Heterogeneous teams and quorum sensing 7

3 The Model **8**

3.1 Swarm model 8

3.2 Group metrics 10

3.3 Group types 11

3.4 Simulating the model 12

4 Model Analysis **13**

4.1 Hysteresis 13

4.2 Tipping point 15

4.3 Analysis of the torus attractor 16

4.3.1 Analysis of attraction-only dynamics 16

4.3.2 Analysis of attraction and orientation dynamics 21

4.4 Analysis of the flock attractor 25
4.5 Summary 27

5 Measuring Human-Swarm Interactions 28

5.1 Interaction effort 28
5.2 Observation effort 29

6 Characterization of Collective Behavior 30

6.1 Global indicators of collective behavior 30
6.2 Local classification 32
6.3 Bayesian classification of swarm group type 33
6.4 Group type classification data sets 34
6.4.1 Training data 34
6.4.2 Test data 34
6.5 Description of classifiers 35
6.6 Classification using number of neighbors 35
6.6.1 Naive Bayes formulation 37
6.6.2 Classification accuracy and observation effort 37
6.6.3 Error analysis 41
6.7 Classification using angular velocity 42
6.7.1 Absolute angular velocity 44
6.7.2 Classification accuracy and observation effort 45
6.8 Combining number of neighbors and absolute angular velocity 48
6.9 Classifying group type and rotation 48
6.9.1 Classification using only angular velocity 49
6.10 Combining angular velocity and number of neighbors to classify group type and rotation 51
6.11 Neglect benevolence for group type classification 54

6.12 Classification during transient behavior 60
6.13 Potential scalability to larger group sizes 62
6.14 Summary 62

7 Estimation of Collective State **64**

7.1 Collective state estimation for a flock 64
7.1.1 Centroid estimation 65
7.1.2 Heading estimation 65
7.1.3 Error analysis for sample group heading 67
7.2 Estimation effort for a flock 68
7.2.1 Experiment 69
7.2.2 Results 70
7.3 Collective state estimation for a torus 74
7.3.1 Group rotation 74
7.3.2 Centroid estimation 74
7.3.3 Torus radius 76
7.3.4 Error analysis 76
7.4 Estimation effort for a torus 77
7.5 Summary 78

8 Classifying the Collective Behavior of Other Models **80**

8.1 Couzin's Model 80
8.1.1 Tipping points 81
8.1.2 Classification features 83
8.1.3 Torus and flock classification 84
8.1.4 Swarm, torus, and flock classification 88
8.2 Physicomimetics model 91
8.3 Discussion 93

9 Quorum Sensing **94**

9.1 Stakeholders 94

9.2 Type-aware agents 96

9.3 Switching between attractors under human influence 97

9.3.1 Switching from flock to torus 100

9.3.2 Switching from torus to flock 100

9.4 Scalability of the quorum sensing model compared to the stakeholder model 101

9.5 Limited vulnerability 106

9.5.1 Switching from torus to flock 107

9.5.2 Switching from flock to torus 109

9.6 Summary 111

10 Conclusions and Future Work **113**

10.1 Future work 114

A Classification using number of neighbors and absolute angular velocity **116**

B Results for classification error analysis **118**

B.1 Error analysis for classification using only the number of neighbors 118

B.2 Error analysis for classificiation using only absolute angular velocity 120

B.3 Error analysis for classification using angular velocity 122

B.4 Error analysis for classification during transient behaviors 123

B.5 Error analysis for two-group classification in Couzin's model 124

B.6 Error analysis for three-group classification for Couzin's model 125

B.7 Tables of classifier accuracies 127

C Supplementary Results for Quorum Sensing **130**

C.1 Scalability 130

C.2 Sensitivity analysis 130

C.3 Effects of heterogeneous parameters 138

D Maximum Speed of a Torus **141**

E Discrete-Time Approximation Error for Attraction-Only Dynamics **144**

E.1 Approximation error 145

E.2 Error 149

E.2.1 Analysis of results 149

E.2.2 Controlling the radius of a swarm through the gain 150

E.3 Simulation results and validation 150

E.4 Dependence on initial conditions 151

E.5 Discussion 152

References **153**

List of Figures

2.1 100 agents in the three different group types formed by our model. 6

4.1 The average change in group momentum and polarization as the radius of orientation is increased and decreased. 14

4.2 Probability of a swarm forming a flock or a torus as a function of radius of orientation. 16

4.3 Coordinates for an agent (blue) attracted toward the centroid (red). 18

4.4 A cyclic group formed by an attraction only swarm. 21

4.5 The trajectories of five simulations run for 100,000 seconds each with initial conditions set to form a flock starting close to the origin. 26

4.6 The distribution of $\Delta\theta_g$ over a single 100,000-second simulation with initial conditions set to form a flock. 27

6.1 The average Fiedler eigenvalue over 100 simulations of a flock and a torus. . 31

6.2 Bayesian network for group type classification using samples of local information from Z agents. 33

6.3 Distributions over an agent's number of neighbors in torus and a flock. . . . 36

6.4 Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying whether the group type is a torus or flock using the number of neighbors as the only feature. 38

6.5 Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying whether the group type is a torus or flock using the number of neighbors as the only feature. 39
6.6 Observation effort for group type classification over different accuracy thresholds μ when using the number of neighbors sampled from Z agents. 40
6.7 An agent sampled (the green disc) from the long tail of flock has fewer neighbors (the 2 red triangles) than those in the main clump of the flock resulting in the misclassification of the flock as a torus. 42
6.8 Distributions over an agent's angular velocity (radians/second) in a clockwise torus, counterclockwise torus, and a flock. 43
6.9 Distributions over an agent's absolute angular velocity (radians/second) in a torus and a flock. 44
6.10 Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying whether the group type is a torus or flock using absolute angular velocity as the only feature. 46
6.11 Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying whether the group type is a torus or flock using absolute angular velocity as the only feature. 46
6.12 Observation effort for group type classification for different accuracy thresholds when sampling absolute angular velocity from Z. 47
6.13 Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying the group type and rotation using angular velocity as the only feature. 49
6.14 Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying the group type and rotation using angular velocity as the only feature. 50

6.15 Observation effort when classifying group type and rotation using angular velocity sampled from Z agents and for different accuracy thresholds μ. . . . 51
6.16 Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying the group type and rotation using both angular velocity and number of neighbors. 52
6.17 Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying the group type and rotation using both angular velocity and number of neighbors. . . . 52
6.18 Observation effort when classifying group type and rotation using angular velocity and number of neighbors sampled from Z agents for different accuracy thresholds μ. Observation for $\mu = 90\%$ and $\mu = 95\%$ are identical. 53
6.19 Average group angular momentum (m_{group}) and group polarization (p_{group}) for a flock and a torus started from random initial conditions. 57
6.20 Average group elongation over time for a flock and a torus started from random initial conditions. 58
6.21 Average group coverage gap over time for a flock and a torus started from random initial conditions. 58
6.22 Neglect benevolence for classifying between clockwise torus counterclockwise torus and flock using one sample ($OT = 1$) of the number of neighbors and the angular velocity from Z agents. Sample Time represents the simulation time at which the samples were taken, or in other words the time before any samples were taken. 59
6.23 Accuracy for classifying between clockwise torus counterclockwise torus and flock using one sample ($OT = 1$) of the number of neighbors and the angular velocity from Z agents. 61

7.1 Flock centroid error for different values of Z. Results are for 100 replicates of 100 simulation ticks each for a flock formation. Dots represent the average error and error bars represent 95% confidence intervals for the error. $N = 100$ agents. 66

7.2 Mean absolute error (in radians) between the vector-based estimate of flock heading, $\hat{\theta}_g$, and actual flock heading for different numbers of sampled agents. Error bars show 95% confidence intervals. $N = 100$ agents. 67

7.3 Mean absolute error versus ET for estimating c_0 and θ_g for a flock, using linear regression for different values of Z. The error in 7.3(b) in in measured in radians. 71

7.4 Mean absolute error versus Z for estimating c_0 and θ_g for a flock, using linear regression for different values of ET. 72

7.5 Estimation effort to get the estimation error for c_0 and θ_g below the threshold ϵ. 73

7.6 Mean absolute error of fixed-radius estimates and naive estimates over samples from 100 simulations leaving out the first 25 seconds for different values of Z. Results for $N = 100$ agents. 77

7.7 Mean absolute error versus estimation time for estimating the centroid of a torus using samples from Z agents. Results are for $N = 100$ agents. 78

7.8 Estimation effort required to estimate the centroid of a torus using samples from Z agents with an average error less than the threshold ϵ. Results are for $N = 100$ agents. 79

8.1 Probability of Couzin's model producing a swarm, a torus, and a flock for different values of R_o. 82

8.2 The percentage of simulations that converged to either a swarm, torus, or flock. 83

8.3 Distributions over an agent's number of neighbors in a swarm, torus, and flock formed by Couzin's model. 84

8.4 Distributions over an agent's angular velocity (radians/second) in a swarm, torus, and flock formed by Couzin's model. 85

8.5 Classification accuracy for Couzin's model using an agent's number of neighbors and angular velocity as features. Results are for distinguishing between a flock, a clockwise torus, and a counterclockwise torus. 86
8.6 Classification accuracy for Couzin's model using the number of neighbors and the angular velocity sampled as a function of OT. Results are for distinguishing between a flock, a clockwise torus, and a counterclockwise torus. 86
8.7 Observation effort calculated for Couzin's model using the number of neighbors and the angular velocity sampled from Z agents. 87
8.8 Classification accuracy as a function of Z using angular velocity and number of neighbors as features. Results are for distinguishing between a swarm, a clockwise torus, a counterclockwise torus, and a flock. 89
8.9 Classification accuracy as a function of OT using angular velocity and number of neighbors as features. Results are for distinguishing between a swarm, a clockwise torus, a counterclockwise torus, and a flock. 90
8.10 Observation effort required to classify the swarm, clockwise torus, counterclockwise torus, and flock group types in Couzin's model for different classification accuracy thresholds. 90
8.11 Three different group types in a physicomimetics particle swarm model. . . . 92

9.1 Switching from flock to torus using quorum sensing. 99
9.2 Switching from torus to flock using quorum sensing. 102
9.3 Switch from a torus to a flock using the stakeholder model and the quorum sensing model. 103
9.4 Switch from a flock to torus using the stakeholder model and the quorum sensing model. 104

9.5 Minimum number of stakeholders, M, needed to switch from torus to flock where stakeholders are led by orientation (t2fo) and to switch from flock to torus where stakeholders are led by attraction (f2ta). Results compare the stakeholder model with the quorum sensing model. 105

9.6 Minimum percentage of total agents that need to be stakeholders to switch from torus to flock where the stakeholders are led by orientation (t2fo) and to switch from flock to torus where the stakeholders are led by attraction (f2ta). Results compare the stakeholder model with the quorum sensing model. . . . 106

9.7 Switch from a torus to a flock using only stakeholders and using quorum sensing with Q=0–6. The M stakeholders are led by orientation. N = 100. . 108

9.8 Switch from a flock to a torus using only stakeholders and using quorum sensing with Q = 0–4. The M stakeholders are led by attraction. N=100. . . 110

9.9 Group polarization over time for one simulation of switching between flock to torus using the stakeholder model and using the quorum sensing model. Human influence is applied at $t = 25$ and released at $t = 225$. 112

A.1 Probability of group type detection as the number of agents sampled, Z, increases for different observation times, OT. For N = 100 agents. 116

A.2 Probability of group type detection as observation time, OT, increases for different numbers of agents sampled, Z. For N = 100 agents. 117

A.3 Observation effort over Z and for different accuracy thrsholds μ. Using $Z = 1$ never resulted in the desired accuracy so the observation effort is infinity. The results for $\mu = 90\%$ aand $\mu = 95\%$ were identical. Results are for N = 100 agents. 117

B.1 An agent sampled (the green disc) from the edge of the flock has fewer neighbors (the 16 red triangles) than those in the main clump of the flock resulting in the misclassification of the flock as a torus. 120

B.2 Torus that was misclassified as a flock. The agent has 17 neighbors. Figure 6.3 shows that the probability of an agent having 17 neighbors is more likely for a flock than for a torus resulting in a misclassification of the torus using one sample from one agent. 121

C.1 Switch from a torus to a flock using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by orientation. N=200. 131

C.2 Switch from a flock to torus using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by attraction. N=200. 132

C.3 Switch from a torus to a flock using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by orientation. N=300. 133

C.4 Switch from a flock to torus using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by attraction. N=300. 134

C.5 Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $Q = 0$ and $0.1 \leq \alpha^{max} \leq 1$. 135

C.6 Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $\rho = 0.7$, $Q = 0, 1, 2$, $\alpha^{max} = 0.6$, and $\alpha^{min} = 0$. 136

C.7 Average interaction time to switch from torus to flock for stakeholder model, and quorum sensing model for $Q = 0$ and $0.1 \leq \alpha^{max} \leq 1$. 136

C.8 Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $Q = 2$, $\alpha^{max} = 0.6$, and $0.1 \leq \alpha^{min} \leq 0.5$. . . 137

C.9 Average interaction time to switch from torus to flock for stakeholder model, and quorum sensing model for $Q = 2$, $\alpha^{max} = 0.8$, and $0 \leq \alpha^{min} \leq 0.7$. . . . 138

C.10 Switch from flock to torus using quorum sensing with homogeneous and heterogeneous parameters. 139

C.11 Switch from torus to flock using quorum sensing with homogeneous and heterogeneous parameters. 140

D.1 Idealized trajectory of an agent represented as a cycloid generated by a circle of radius r that has rotated through the angle θ. 142

E.1 N-gonal approximation of a circle 146

E.2 Error in heading ($\tilde{\theta}$) for discrete-time simulation 148

E.3 Trajectories of a centroid for 5 agents with $\Delta t = .5$. 152

List of Tables

6.1 The sample mean and standard deviation for the distributions of number of neighbors over natural group types. 35

6.2 Values of Z that minimized observation effort for μ = 90, 95, and 100% accuracy on the test set using only the number of neighbors as a feature. . . 40

6.3 The sample mean and standard deviation for the distributions of angular velocity shown in Figure 6.8 42

6.4 The sample mean and variance of the absolute angular velocity of agents in a flock and torus. 44

6.5 Values of Z that minimized observation effort (OE) for $\mu = 90$, 95, and 100% accuracy on the test set using only absolute angular velocity as a feature. . . 47

6.6 Values of Z that minimized observation effort (OE) for $\mu = 90$, 95, and 100% accuracy on the test set using only angular velocity as a feature. 50

6.7 Values of Z that minimized observation effort (OE) for $\mu = 90$, 95, and 100% accuracy on the test set using angular velocity and the number of neighbors as features. 53

6.8 Statistics for group elongation and coverage gap for a fully formed flock and torus. 55

6.9 Statistics for the time required for a flock or a torus to fully form when using either the group elongation or the group coverage gap as a measure of how well the group has formed. 56

6.10 The sample mean and standard deviation for the distributions of number of neighbors as the size of the swarm, N, increases. 62
6.11 The sample mean and standard deviation for the distributions of angular velocity as the size of the swarm, N , increases. 63

8.1 Values of Z that minimized observation effort in Figure 8.7 for $\mu = 90$, 95, and 100% accuracy on the test set. 87
8.2 Values of Z that minimized observation effort in Figure 8.10 for $\mu = 90$, 95, and 100% accuracy on the test set. 89

B.1 Confusion matrix for $Z = 1$ and $OT = 1$ using only the number of neighbors as a feature. 118
B.2 Confusion matrix for $Z = 2$ and $OT = 1$ using only the number of neighbors as a feature. 119
B.3 Confusion matrix for $Z = 1$ and $OT = 2$ using only the number of neighbors as a feature. 119
B.4 Confusion matrix for $Z = 1$ and $OT = 1$ using only absolute angular velocity as a feature. 120
B.5 Confusion matrix for $Z = 2$ and $OT = 1$ using only absolute angular velocity as a feature. 121
B.6 Confusion matrix for $Z = 1$ and $OT = 2$ using only absolute angular velocity as a feature. 122
B.7 Confusion matrix for $Z = 5$ and $OT = 5$ using only absolute angular velocity as a feature. 122
B.8 Confusion matrix for $Z = 1$ and $OT = 1$ using only angular velocity to classify group type and rotation. 122
B.9 Confusion matrix for $Z = 2$ and $OT = 1$ using only angular velocity to classify group type and rotation. 123

B.10 Confusion matrix for $Z = 1$ and $OT = 2$ using only angular velocity to classify group type and rotation. 123
B.11 Confusion matrix for $Z = 5$ and $OT = 5$ using only angular velocity to classify group type and rotation. 123
B.12 Confusion matrix for group type classification when sampling angular velocity and number of neighbors with $Z = 1$ and $OT = 1$ after 10 seconds of simulation. 124
B.13 Confusion matrix for group type classification when sampling angular velocity and number of neighbors with $Z = 5$ and $OT = 1$ after 30 seconds of simulation. 124
B.14 Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 1$ and $OT = 1$. . 124
B.15 Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 5$ and $OT = 5$. . 125
B.16 Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 15$ and $OT = 15$. 125
B.17 Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 1$ and $OT = 1$. . . . 126
B.18 Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 5$ and $OT = 5$. . . . 126
B.19 Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 15$ and $OT = 15$. . . 126
B.20 Percent accuracy for using only number of neighbors to predict the unoriented group type of a swarm. 127
B.21 Percent accuracy for using only angular velocity to predict the unoriented group type of a swarm. 127
B.22 Percent accuracy for using only angular velocity to predict the group type and, in the case of the torus, orientation of the swarm. 128

B.23 Percent accuracy for using both angular velocity and number of neighbors to predict the group type and, in the case of the torus, orientation of the swarm. 128
B.24 Percent accuracy for using only number of neighbors to predict the oriented group type for Couzin's model. 129

E.1 Comparison of simulated results with theoretical results. The last row shows the values for the continuous case. 151

Chapter 1

Introduction

In nature, multi-agent groups achieve remarkable feats: ant colonies find the shortest path to distant food sources, flocks of birds swoop and dive in perfect formation, and colonies of bees efficiently evaluate potential nest sites and decide on the best location. Despite the limited intelligence and abilities of each individual agent, a collective intelligence is achieved that is greater than the sum of its parts [1]. Applying principles from biological systems to create bio-inspired robot teams leverages the decentralized behaviors found in nature to create scalable and robust teams that are amenable to human interaction [2].

In many situations, such as search and rescue, military operations, and space exploration, it is desirable to have a robust robot team that is amenable to human interaction. Bio-inspired robot swarms are typically characterized by a large number of agents with limited abilities and are ideally suited for accomplishing dangerous or uncertain tasks where communication is limited and agent failures are likely. As these bio-inspired teams grow in size, bandwidth and time constraints make it increasingly difficult for a human to interact with every individual agent. Thus, there is a need for scalable human-swarm interactions that allow a human to monitor and influence the collective behavior of the swarm through interactions with a subset of the swarm. Additionally, because many swarm applications involve dangerous or uncertain environments, we desire swarms that have limited vulnerability to agent failure—we want the collective behavior of the swarm to be unresponsive to small numbers of agent failures.

This thesis will focus on two major problems that impede scalable human interaction with large bio-inspired robot swarms. First, how can we accurately detect and estimate the collective state of a robot swarm using limited information from a subset of the swarm? Second, how can we increase the scalability of human influence over the collective behavior of a swarm while limiting the vulnerability of the swarm to agent failures?

Much of the current research involving bio-inspired teams has either involved centralized control through global information or has assumed that agents can explicitly communicate with one another [3] [4] [5]. Some exceptions are Couzin *et al.* [6] and Conradt *et al.* [7] who show that that a limited number of individuals with a preferred direction can influence the direction of collective movement in a swarm with no explicit communication. Couzin's work [8] [6] has been generalized by our lab (BYU HCMI Lab) to provide methods for leading multiple group types and switching between distinct group behaviors [9]. However, our previous work assumes perfect knowledge of the collective behavior of the swarm.

This thesis demonstrates the use of a Bayesian network to detect the collective behavior of a robot swarm using information from only a limited number of agents. We assume that no explicit communication occurs between agents. To simulate swarming behaviors we use a model of robot swarming behavior previously developed by our lab [9]. This model has two fundamental attractors: a flock and a torus. Using local information from a subset of agents, we use a naive Bayes classifier to accurately detect whether the group is in one of these two attractors by using a small sample of agent angular velocities and local agent densities. We also use local information from a subset of the agents to estimate and predict the collective centroid and heading of the swarm. In addition, we argue that the idea of estimating global properties of a system from local observations is applicable to other swarm systems by examining group type classification for a model developed by Couzin *et al.* [8] and discussing how these ideas could also be applied to a physicomimetics model [10].

Finally, this thesis demonstrates that adding a biologically inspired quorum sensing mechanism to a swarm increases the scalability of human-swarm interactions and limits the

vulnerability of the swarm to agent failures. By comparing the performance of a swarm using quorum sensing with the performance of a swarm without quorum sensing we show that quorum sensing reduces the number and duration of human interactions needed to influence the collective behavior of a swarm and allows human-swarm interactions to scale better to larger swarm sizes. Additionally, we demonstrate that the use of quorum sensing reduces the vulnerability of the swarm to agent failures by limiting the swarm's responsiveness if less than a certain threshold of agents change their behavior.

Chapter 2

Related Work

2.1 Models of swarming behavior

Bio-inspired swarm models have been explored by researchers in a wide variety of fields including computer science, engineering, physics, and biology. These models are typically capable of either flocking [3, 5, 11, 12] or cyclic behavior [13, 14], and in some cases can exhibit multiple group behaviors depending on the model parameters used [8, 15, 16].

Couzin *et al.* [6] and Conradt *et al.* [7] explore leading a flock with a small number of informed agents. Couzin *et al.* show that their method of leading a flock scales well as group size increases, but do not consider leading a torus or changing group types.

In the controls community much research focuses on consensus protocols [4] for flocking, but we have not found any research involving switching between attractors without communication or centralized control. Olfati-Saber [3] uses global information and communication with local neighbors to form a robust flock and proves that a leader agent can lead the group through global information. Su *et al.* [17] extend Olfati-Sabers work by eliminating the need for global information and show that only a subset of informed individuals are needed to influence the flock to move with a desired velocity.

Some work has been done with communication-free flocking, but this work typically creates flocks that can be controlled by one or a few agents, which makes the flock vulnerable to leader failure. Gervasi and Prencipe [18] study distributed coordination and control without any communication or shared reference frame, but require that all agents can identify the leader agent and can be efficiently controlled by a single agent. Jadbabaie [5] provides

mathematical results on the convergence of the Vicsek flocking model [12] to a single group direction as well as convergence conditions for the group to converge in the limit to a single leader's direction. LeBlanc [19] provides a consensus protocol that requires limited communication, but provides conditions for reaching consensus in the presence of adversaries.

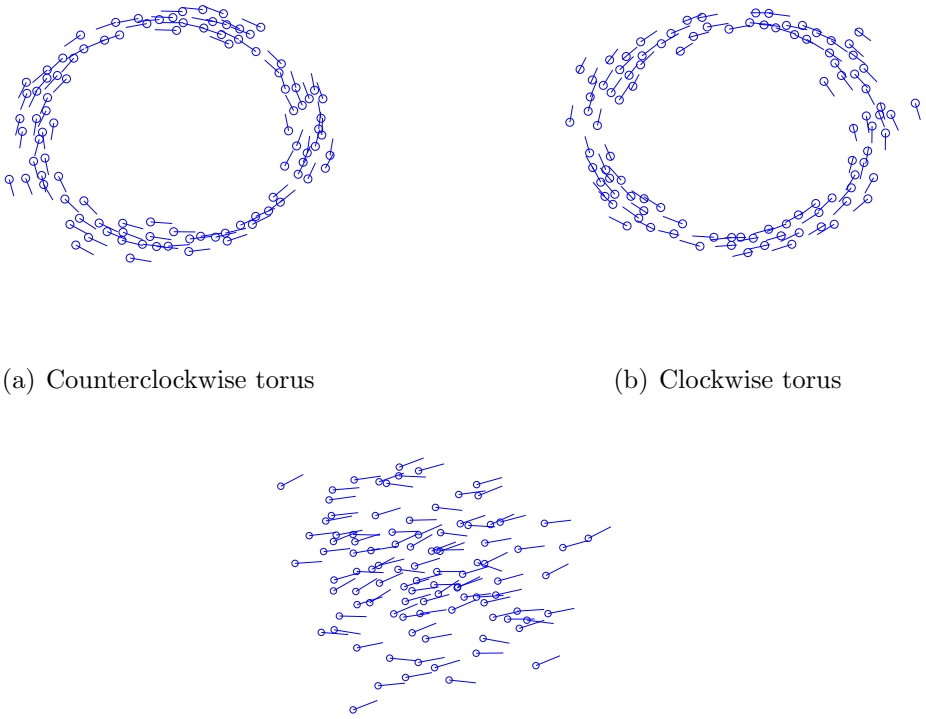
This thesis builds on previous work done in the BYU-HCMI lab. We focus on a simple swarm model[1] we have developed [9]. In this model, agents react to neighbors within three different zones: repulsion, orientation, and attraction. There is no explicit communication—agents only react to visual cues from nearby agents. Each agent follows three basic rules: (1) avoid neighbors within a certain repulsion zone, (2) move in the same direction as neighbors in a larger orientation zone, and (3) stay close to neighbors beyond the orientation zone. From these three simple rules the model produces two fundamental attractors: a moving flock group type and a stationary rotating torus group type which may rotate either clockwise or counterclockwise. These group types are shown in Figure 2.1.

Previous work in our lab shows that flock and torus group types can be led by a subset of informed individuals and that human influence over at least 30% of the swarm allows human-influenced switches between group types [9]. We will use these results as a baseline when we analyze the performance of our proposed swarm model with quorum sensing.

2.2 Types of swarm behavior

Most research on swarm behaviors focuses on flocking. However, there are other types of swarm behavior seen in the literature. Couzin's model [8] has four group types: swarm, torus, dynamic parallel group, and highly parallel group, where the last two group types are simply two different flavors of flocking. Strömbom [15] demonstrates that attractive forces between agents is sufficient to form swarms, flocks, and mills (torus-like formations where agents do not all rotate in the same direction). Strömbom also shows that adding a blind spot creates two additional group types: a torus and an interweaving chain-like structure. Romero *et al.*

[1] This model is explained in detail in Section 3.1



(a) Counterclockwise torus

(b) Clockwise torus

(c) Flock

Figure 2.1: 100 agents in the three different group types formed by our model. The agents' headings are represented by straight lines emanating from the center of each agent.

[16] present a swarm model that produces a swarm, a torus, a flock, and a flock that rotates around a stationary point.

If we restrict our attention to models without centralized control, global information, or explicit inter-agent communication, the group types mentioned above are the only group types we have found in the literature that emerge from swarm models capable of exhibiting multiple behaviors. This thesis focuses on the two group types we feel could be most realistically applied to actual robots: the torus and the flock; other group types are interesting, but often result agent trajectories that would likely result in collisions between actual robots. Because of the large void in the center, we hypothesize that a torus could be used for perimeter monitoring or convoy protection. Because the flock group type provides a more agile and

mobile group type, we hypothesize that a flock could be used for quickly moving a swarm from one location to another.

2.3 Detection and estimation of collective behavior

This thesis uses samples of local information from individual agents to detect and estimate the collective behavior of a swarm. Other researchers have investigated similar topics, but have generally relied only on positional data. Wirz *et al.* [20] propose an approach for detecting collective patterns in crowds using body-worn sensors. Sadilek and Kautz [21] consider the task of inferring high-level human interactions and intentions using location information from noisy GPS data. Laube *et al.* [22] propose an algorithm for the decentralized detection of flocking in mobile networks through the exchange of information tokens. Gudmundsson *et al.* develop efficient approximation algorithms for detecting flocking, leadership, and convergence using moving point object data [23, 24]. Eagle *et al.* [25–27] show how data from mobile phones can be used to infer and analyze social network structures and group behaviors. Tang *et al.* [28] explore scalable learning of collective behaviors in social networks using edge clustering schemes.

2.4 Heterogeneous teams and quorum sensing

Several researchers have studied the effects of heterogeneity and quorum sensing on multi-agent interactions. Wu [29, 30] investigates the importance and role of heterogeneity in robot teams and shows that increased agent variation in task allocation problems can increase stability. Kumar [31] uses mathematical models of bio-inspired foraging tasks and quorum sensing to develop control strategies for redistributing agents among multiple sites but does not investigate human influence over the agent behaviors. This thesis will investigate the benefits of adding quorum sensing to a bio-inspired swarm in terms of scalability and limited vulnerability.

Chapter 3

The Model

3.1 Swarm model

The experiments and simulations in this thesis will build upon on a model of swarming that we have produced in our lab [9]. We have chosen to focus on this model because it can form both flock and torus group types, is similar to many biological models of swarming behavior, and has dynamics similar to those of actual robots. Our model consists of a set of N agents with nonholonomic dynamics for agent i given by

$$\begin{aligned} \dot{x_i} &= s \cdot \cos(\theta_i) \\ \dot{y_i} &= s \cdot \sin(\theta_i) \\ \dot{\theta}_i &= \omega_i \end{aligned} \tag{3.1}$$

where $[x_i, y_i]^T \in \mathbb{R}^2$ is the agent's position, $\theta_i \in [-\pi, \pi]$ is the agent's angular heading, s is the constant agent speed, and w_i is the agent's angular velocity. For simplicity we define

$$v_i = [\cos(\theta_i), \sin(\theta_i)]^T \tag{3.2}$$

$$c_i = [x, y]^T. \tag{3.3}$$

Further, let $A(t) = a_{ij}(t)$ denote the sensory adjacency matrix where $a_{ij}(t) = 1$ means that agent j is visible to agent i at time t. Each $a_{ij}(t)$ is determined at time t according to

a Bernoulli random variable with parameter $p_{ij}(t) = \min\left(1, 1/d_{ij}(t)\right)$ where $d_{ij}(t)$ is the Euclidean distance between agents i and j at time t.

Agents react to neighbors within three different zones: repulsion, orientation, and attraction. The neighbors in these zones are determined by

$$n_i^r = \{j : \|c_i - c_j\| \leq R_r, a_{ij} = 1\} \tag{3.4}$$

$$n_i^o = \{j : \|c_i - c_j\| \leq R_o, a_{ij} = 1\} \tag{3.5}$$

$$n_i^a = \{j : a_{ij} = 1\} \tag{3.6}$$

where n_i^r, n_i^o, and n_i^a are the sets of agent i's neighbors in the regions of repulsion, orientation, and attraction, respectively. Throughout this thesis, $\|\cdot\|$ is used to denote the standard Euclidean norm. The parameters R_r and R_o are the associated radii of repulsion and orientation. The angular velocity ω_i is determined by first computing the repulsion, orientation, and attraction vectors

$$u_i^r = -\sum_{n_i^r} \frac{c_j - c_i}{\|c_j - c_i\|^2} \tag{3.7}$$

$$u_i^o = \frac{v_i + \sum_{n_i^o} v_j}{\|v_i + \sum_{n_i^o} v_j\|} \tag{3.8}$$

$$u_i^a = \frac{\sum_{n_i^a}(c_j - c_i)}{\|\sum_{n_i^a}(c_j - c_i)\|}. \tag{3.9}$$

Next, the desired heading vector u_i is computed as $u_i = u_i^r + u_i^o + u_i^a$. Finally, angular velocity, ω_i, is computed as

$$\omega_i = k(\text{atan2}(u_i^y, u_i^x) - \theta_i) \tag{3.10}$$

where k is a positive gain and $\text{atan2}(u_i^y, u_i^x)$ is the two argument variation of the arctangent that places the angle in the correct quadrant by considering the signs of the y and x

components of u_i. Because we limit $(\text{atan2}(u_i^y, u_i^x) - \theta_i)$ to the interval $[-\pi, \pi]$, the magnitude of ω_i is bounded by $k\pi$.

3.2 Group metrics

In order to define the two different attractors of our model we use two metrics of group behavior described in [8], namely, group angular momentum, m_{group}, and group polarization, p_{group}. Group angular momentum is a measure of the degree of rotation of the group about the group centroid and is calculated as

$$m_{group}(t) = \frac{1}{N} \left| \sum_{i=1}^{N} \det([r_{ic}(t)\ v_i(t)]) \right|. \tag{3.11}$$

The vector $r_{ic}(t)$ is a unit vector pointing from the group centroid to the position of agent i and is given by

$$r_{ic}(t) = \frac{c_i(t) - c_g(t)}{\|c_i(t) - c_g(t)\|} \tag{3.12}$$

$$c_g(t) = \frac{1}{N} \sum_{i=1}^{N} c_i(t) \tag{3.13}$$

where $c_g(t)$ is the group centroid. The term $\det([r_{ic}(t)\ v_i(t)])$ is the determinant of the 2×2 matrix $[r_{ic}(t)\ c_i(t)]$ with columns $r_{ic}(t)$ and $v_i(t)$ and is a two-dimensional analogue of the cross product. The group angular momentum is a value between 0 and 1. This can be seen by examining $\det([a\ b])$ where $a, b, \in \mathbb{R}^2$ and the vectors a and b are both unit vectors. Let ϕ be the angle between them such that $a = R(\phi)b$ where $R(\phi)$ is the rotation matrix

$$R(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}. \tag{3.14}$$

Thus,

$$\begin{aligned} \det([a\ b]) &= a_1 b_2 - a_2 b_1 && (3.15) \\ &= a_1(a_1 \sin\phi + a_2 \cos\phi) - a_2(a_1 \cos\phi - a_2 \sin\phi) && (3.16) \\ &= \sin\phi(a_1^2 + a_2^2) && (3.17) \end{aligned}$$

The term $a_1^2 + a_2^2 = 1$ since a is a unit vector, so we see that $-1 \leq \det([a\ b]) \leq 1$ with maximum and minimum values when $\phi = \pi/2$ and $\phi - \pi/2$, respectively. Thus, the m_{group} of a swarm reaches a maximum value of 1 if all the agents are rotating around the group centroid in the same direction with v_i orthogonal to r_{ic} and a minimum value of 0 when the agents are aligned in the same direction and evenly spaced around the centroid or when half of the agents rotate clockwise and the other half rotate counterclockwise around the group centroid.

Group polarization measures the degree of alignment among individuals within the group and is calculated as

$$p_{group}(t) = \frac{1}{N} \left| \sum_{i=1}^{N} v_i(t) \right|. \tag{3.18}$$

The p_{group} of a swarm reaches a maximum value of 1 when all the agents have the same heading and has a minimum value of 0 when agent headings are evenly distributed in the interval $[-\pi, \pi]$.

3.3 Group types

Our model produces two group types: a torus and a flock. Snap shots of these group types are shown on page 7 in Figure 2.1. A torus is characterized by p_{group} close to 0, m_{group} close to 1, and a relatively stationary group centroid. A flock is characterized by p_{group} close to 1,

m_{group} close to 0, and a moving centroid. Specifically, we define the group type, $type_g$, as

$$type_g = \begin{cases} \text{torus}, & \text{if } m_{group} > 0.75 \text{ and } p_{group} < 0.25 \\ \text{flock}, & \text{if } m_{group} < 0.25 \text{ and } p_{group} > 0.75. \end{cases} \tag{3.19}$$

3.4 Simulating the model

Simulations were run using the following discrete-time approximation of the dynamics in Equation (3.1)

$$\begin{aligned} \theta_i(t+1) &= \theta_i(t) + \omega_i(t) \cdot \Delta t \\ x_i(t+1) &= x_i(t) + s \cdot \cos(\theta_i(t+1)) \cdot \Delta t \\ y_i(t+1) &= y_i(t) + s \cdot \sin(\theta_i(t+1)) \cdot \Delta t \end{aligned} \tag{3.20}$$

with simulation time step $\Delta t = 0.1$ seconds. This time step is in the range of update times (approximately 1-100 Hz) for many common robotic sensors (see Appendix H of [32]) and leads to a tractable simulation time.

Chapter 4

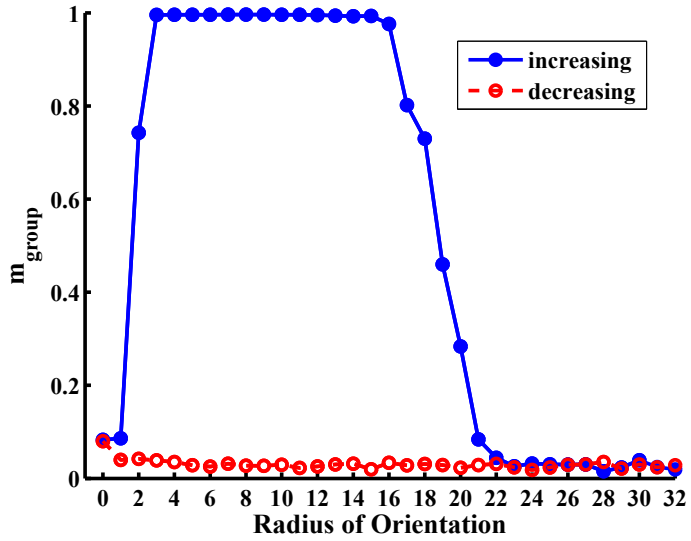
Model Analysis

Before beginning the main body of this thesis, we first present several empirical and theoretical results regarding our swarm model that will be used in subsequent sections.

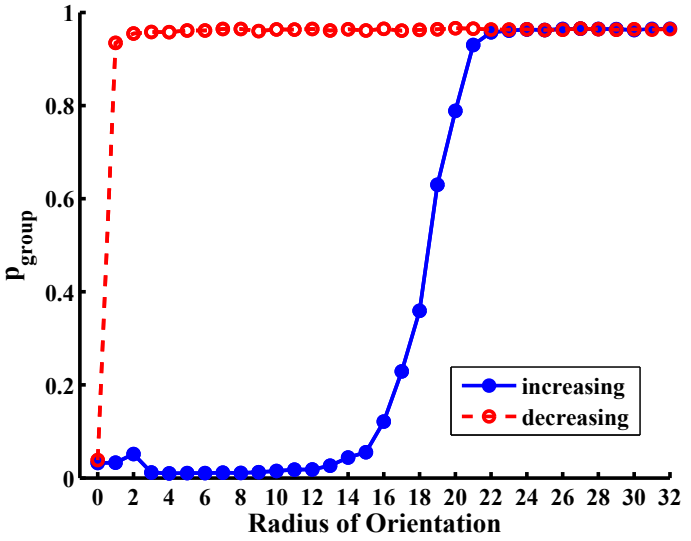
4.1 Hysteresis

To investigate the collective memory of our swarm model, we followed the methodology used by Couzin *et al.* [8]. We fixed the radius of repulsion, $R_r = 1$, and experimented with slowly increasing and decreasing the radius of orientation, R_o. The parameter R_o was incremented or decremented by 1 unit every 1000 time steps and the results were averaged over 15 replicates. Model parameters other than R_o were $N = 100$, $k = 0.5$, and $s = 5$. The results shown in Figure 4.1 consist of the average m_{group} and p_{group} calculated one time step before the radius of orientation was incremented or decremented. As can be seen, the group behavior depends on the previous history of the group, even though the individual agents have no explicit knowledge of what that history is. Interestingly, when decreasing the radius of orientation, the flock does not ever switch to a torus but simply remains a flock until $R_o = 0$ at which point the flock turns into an unoriented cyclic group.

From these results we see that the hysteresis in our model precludes switching between the flock and torus group types by increasing or decreasing the parameter R_o—the only way to go from a flock to a torus is to change the group to a cyclic group and then increase R_o to get a torus. Because we are interested in human-influenced switching between group types, but do not want to switch group types by drastically changing global parameters, we



(a)



(b)

Figure 4.1: The average change in group momentum and polarization as the radius of orientation is increased and decreased.

ideally want to find a sort of "tipping point" between the torus and flock group types, i.e., a set of parameters that can exhibit both the flock and torus group types equally well. This would allow a human to influence the collective behavior of the swarm without needing to slide up and down the hysteresis curve and without worrying about globally broadcasting parameter changes or using some kind of consensus protocol to have agents change their radius of orientation.

4.2 Tipping point

We desire to be able to switch between the torus and flock attractors without changing model parameters. To determine parameter values that allow both group types to emerge we ran a series of simulations using $N = 100$, $k = .5$, $s = 5$, $R_r = 1$ and varied the the radius of orientation. Simulations were run for 200 seconds with a time step size of $\Delta t = 0.1$. The radius of orientation was varied from 0 to 30 in 1-unit increments. One hundred simulations were performed for each value of R_o. For each iteration, agents were given random initial positions uniformly distributed over a 10×10 square centered at the origin. Agents were also given random initial headings.

The percentage of trials that converged to a torus and to a flock were calculated for each value of R_o and are shown in Figure 4.2. As can be seen in the figure, the value $R_o = 8$ resulted in an approximately equal proportion of torus and flock group types. Figure 2.1 (shown on page 7) shows a counterclockwise torus, a clockwise torus, and a flock formed with the parameter values listed above and $R_o = 8$.

Using $R_o = 8$ allows distinct group types to emerge from the same model parameters. The only differences between simulations were the initial headings and initial positions of the agents at the beginning of each simulation. This provides evidence that we can change the collective behavior of the swarm by influencing some of the agents in the collective to change their position and heading without requiring parameter rebroadcasting or communication

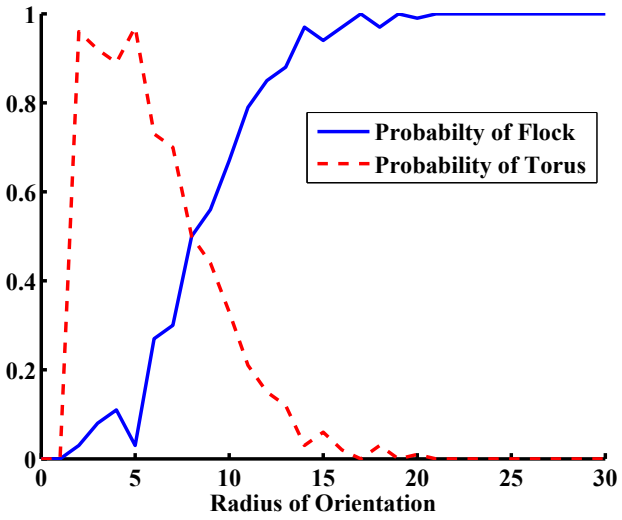


Figure 4.2: Probability of a swarm forming a flock or a torus as a function of radius of orientation.

between the agents. For the remainder of this thesis we use the parameters $N = 100$, $k = 0.5$, $s = 5$, $R_r = 1$, and $R_o = 8$ unless otherwise specified.

4.3 Analysis of the torus attractor

Our model produces two fundamental group types, a flock and a torus. In this section, we provide an argument that the torus group type is a formal attractor of our model. We first consider a system based only on attraction ($u_i = u_i^a$) and prove it converges to a stable cycle. Next we consider a system with both attraction and orientation dynamics ($u_i = u_i^a + u_i^o$) and derive an equation for the radius of a torus formed by attraction and orientation dynamics.

4.3.1 Analysis of attraction-only dynamics

We assume a complete agent topology ($a_{ij} = 1 \; \forall \; i \neq j$). Using this assumption the desired direction of travel for agent i is

$$u_i = \frac{\sum_{j \neq i}^{N} (c_j - c_i)}{\| \sum_{j \neq i}^{N} (c_j - c_i) \|}. \tag{4.1}$$

Because the denominator is a positive scalar, it does not affect the direction of u_i and can be ignored because the agent dynamics only depend on the direction of u_i. Thus, we can write u_i as

$$\begin{aligned} u_i &= \sum_{j \neq i}^{N} (c_j - c_i) && (4.2) \\ &= \sum_{j=1}^{N} (c_j - c_i) && (4.3) \\ &= \left(\sum_{j=1}^{N} c_j \right) - N c_i. && (4.4) \end{aligned}$$

Multiplying by any positive scalar will preserve direction so we multiply u_i by $1/N$ to get

$$u_i = c_g - c_i \tag{4.5}$$

where c_g is the group centroid. Therefore, for each agent i, u_i points toward the group centroid.

Change of Variables

Consider an agent that is attracted toward the group centroid c_g. The agent's dynamics are given by

$$\begin{aligned} \dot{x} &= s \cdot \cos\theta \\ \dot{y} &= s \cdot \sin\theta \\ \dot{\theta} &= \omega. \end{aligned} \tag{4.6}$$

where $\omega = k\alpha$ and $\alpha = \text{atan2}(u^x, u^y) - \theta$. We want to analyze our model in terms of the agent's distance from the centroid and the angle $\alpha = \text{atan2}(u^x, u^y) - \theta$ which is the difference between the agent's current heading θ and the heading that would take it directly toward the

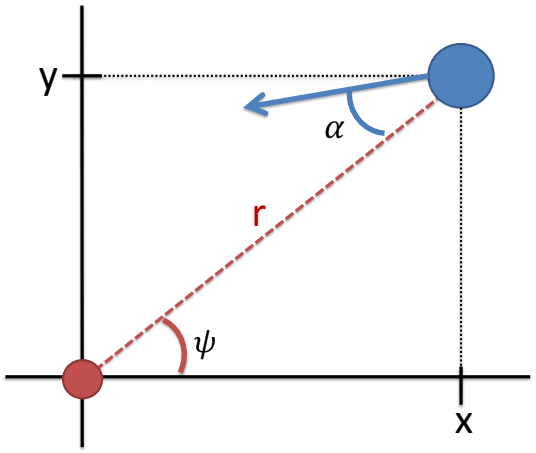


Figure 4.3: Coordinates for an agent (blue) attracted toward the centroid (red).

centroid. We assume a stationary group centroid at the origin and, using a method similar to [13], we perform the change of variables

$$r = \sqrt{x^2 + y^2} \tag{4.7}$$

$$\alpha = \psi + \pi - \theta \tag{4.8}$$

where r is the distance from the group centroid and $\psi = \arctan\left(\frac{y}{x}\right)$. Figure 4.3 shows how these variables relate to each agent. Deriving an equation for $\dot{r}$ we have

$$\begin{aligned} \dot{r} &= \frac{x\dot{x} + y\dot{y}}{\sqrt{x^2 + y^2}} && (4.9)\\ &= \cos\psi\dot{x} + \sin\psi\dot{y} && (4.10)\\ &= s(\cos\psi\cos\theta + \sin\psi\sin\theta) && (4.11)\\ &= -s(\cos\psi\cos(\psi - \alpha) + \sin\psi\sin(\psi - \alpha)). && (4.12) \end{aligned}$$

Using the identity $\cos u \cos v + \sin u \sin v = \cos(u - v)$ we have

$$\dot{r} = -s \cos \alpha. \tag{4.13}$$

We now derive an equation for $\dot{\alpha}$

$$\begin{aligned} \dot{\alpha} &= \frac{d}{dt} \arctan\left(\frac{y}{x}\right) - \dot{\theta} && (4.14)\\ &= \frac{x\dot{y} - y\dot{x}}{x^2 + y^2} - k\alpha && (4.15)\\ &= \frac{s}{r^2}(x \sin\theta - y \cos\theta) - k\alpha && (4.16)\\ &= \frac{s}{r}[\cos\psi \sin(\alpha - \psi) + \sin\psi \cos(\alpha - \psi)] - k\alpha. && (4.17) \end{aligned}$$

Using the identity $\cos u \sin v + \sin u \cos v = \sin(u + v)$ we have

$$\dot{\alpha} = \frac{s}{r} \sin\alpha - k\alpha. \tag{4.18}$$

Thus, we have

$$\begin{aligned} \dot{r} &= -s \cos\alpha && (4.19)\\ \dot{\alpha} &= \frac{1}{r} s \sin\alpha - k\alpha && (4.20) \end{aligned}$$

which describe an agent's dynamics in terms of its distance from the centroid, r, and the desired change in heading, α.

Stability of Equilibrium Points

Setting $\dot{r}$ and $\dot{\alpha}$ equal to 0 and solving for the equilibrium points of (4.19) and (4.20) we get

$$r^{eq} = \frac{2s}{k\pi}, \quad \alpha^{eq} = \pm\frac{\pi}{2} \tag{4.21}$$

where we have restricted α to be in the interval $[-\pi, \pi]$. These two equilibria define a clockwise and counterclockwise orbit about the fixed centroid with radius $r = 2s/k\pi$.

We now investigate the stability of the equilibria through a local linearization of equations (4.19) and (4.20) around the equilibrium points. If we define $f_1 := -s\cos\alpha$ and $f_2 := \frac{s}{r}\sin\alpha - k\alpha$, the Jacobian matrix is

$$\begin{bmatrix} \frac{\partial f_1}{\partial r} & \frac{\partial f_1}{\partial \alpha} \\ \frac{\partial f_2}{\partial r} & \frac{\partial f_2}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} 0 & s\sin\alpha \\ -\frac{s}{r^2}\sin\alpha & \frac{s}{r}\cos\alpha - k \end{bmatrix}. \tag{4.22}$$

Evaluating the Jacobian at $r^{eq} = 2s/k\pi$ and $\alpha^{eq} = \pi/2$ and letting $\omega = k\alpha^{eq}$ we have

$$\begin{bmatrix} 0 & s\sin\alpha \\ -\frac{s}{r^2}\sin\alpha & \frac{s}{r}\cos\alpha - k \end{bmatrix}\Bigg|_{(r^{eq},\alpha^{eq})} = \begin{bmatrix} 0 & s \\ -\frac{k^2\pi^2}{4s} & -k \end{bmatrix}. \tag{4.23}$$

Solving for the eigenvalues we have

$$\lambda = \frac{k}{2}(-1 \pm i\sqrt{\pi^2 - 1}). \tag{4.24}$$

Since $k > 0$, both eigenvalues have negative real parts and nonzero imaginary parts. Therefore the equilibrium point is locally asymptotically stable [33]. Linearizing about the other equilibrium point

$$r = \frac{2s}{k\pi}, \quad \alpha = -\frac{\pi}{2} \tag{4.25}$$

gives the same result.

This indicates that for a stationary group centroid, all agents converge to either a clockwise or counterclockwise orbit about the group centroid with a fixed radius $r = 2s/k\pi$. This behavior is shown in Figure 4.4. Thus we argue that the torus group type is a fundamental attractor of our dynamic system. The addition of orientation and repulsion forces to the

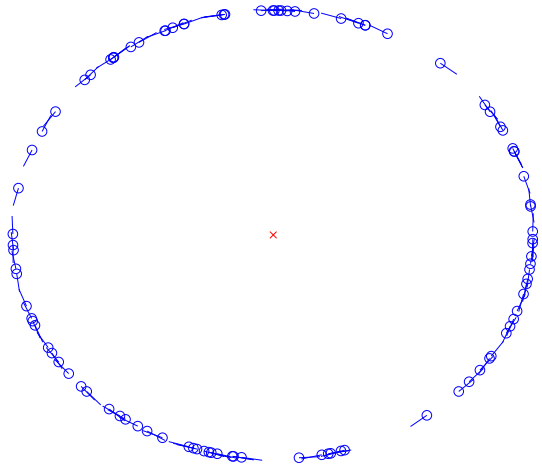


Figure 4.4: A cyclic group formed by an attraction only swarm. The group centroid is marked by an 'x'.

model creates an oriented torus with collision avoidance and also allow the flock group type to form.

4.3.2 Analysis of attraction and orientation dynamics

In the previous section we argued that the torus group type is a stable attractor of our model with attraction-only dynamics. In this section we investigate the radius of a torus with attraction and orientation dynamics, i.e., $u_i = u_i^a + u_i^o$.

Lemma 1. $(\sin\theta + \cos\theta)/(\cos\theta - \sin\theta) = \tan(\pi/4 + \theta)$

Proof. Starting with the right side of the equality to prove, we have

$$\begin{aligned}
\tan(\pi/4+\theta) &= \frac{\sin(\pi/4+\theta)}{\cos(\pi/4+\theta)} && (4.26)\\
&= \frac{\sin(\pi/4)\cos\theta + \cos(\pi/4)\sin\theta}{\cos(\pi/4)\cos\theta - \sin(\pi/4)\sin\theta} && (4.27)\\
&= \frac{\frac{1}{\sqrt{2}}(\cos\theta + \sin\theta)}{\frac{1}{\sqrt{2}}(\cos\theta - \sin\theta)} && (4.28)\\
&= \frac{\sin\theta + \cos\theta}{\cos\theta - \sin\theta} && (4.29)
\end{aligned}$$

where the second line results from using the trigonometric identities:

$$\cos u \cos v \mp \sin u \sin v = \cos(u \pm v)$$

$$\sin u \cos v \pm \cos u \sin v = \sin(u \pm v).$$

□

Lemma 2. $(\sin\theta - \cos\theta)/(\cos\theta + \sin\theta) = \tan(-\pi/4 + \theta)$

Proof. The proof is almost identical to the proof for Lemma 1. □

Theorem 1. *The angular velocity of an agent in a well balanced attraction-orientation torus with a complete topology is one-half the angular velocity of an agent in an attraction-only torus.*

Proof. We have previously shown that the angular velocity of an agent in an attraction-only torus is $\omega = \pm k\pi/2$. When we add orientation, ω becomes

$$\omega = k(\text{atan2}(u) - \theta) = k(\text{atan2}(u^o + u^a) - \theta). \tag{4.30}$$

To simplify this term we assume a well balanced torus, i.e.,

$$u^o = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}. \tag{4.31}$$

This basically means that we assume the agents are nicely spaced around the torus so that when we compute

$$u_i^o = \frac{v_i + \sum_{n_i^o} v_j}{\|v_i + \sum_{n_i^o} v_j\|} \tag{4.32}$$

the velocity vectors of the neighbors cancel out and we are left with $u_i^o = v_i$.

We also assume a complete topology. Thus,

$$\omega = k\left(\text{atan2}\left(\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix} + \frac{c_g - p_i}{\|c_g - p_i\|}\right) - \theta\right). \tag{4.33}$$

Without loss of generality, we assume that $c_g = [0,0]^T$. To simplify the math, we also assume that the angles are such that atan2 = arctan. Therefore, the atan2 term in Equation (4.33) simplifies to

$$\begin{aligned}
\text{atan2}\left(\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix} - \frac{p_i}{\|p_i\|}\right) &= \arctan\left(\frac{\sin\theta - \frac{y}{\sqrt{x^2+y^2}}}{\cos\theta - \frac{x}{\sqrt{x^2+y^2}}}\right) &(4.34)\\
&= \arctan\left(\frac{\sin\theta - \sin\psi}{\cos\theta - \cos\psi}\right) &(4.35)\\
&= \arctan\left(\frac{\sin\theta \mp \cos\theta}{\cos\theta \pm \sin\theta}\right) &(4.36)
\end{aligned}$$

where we have once again used the change of variables $r = \sqrt{x^2+y^2}$, $x = r\cos\psi$, and $y = r\sin\psi$ (see Figure 4.3) to obtain (4.35). To arrive at (4.36) we use that fact that as an agent moves along its circular trajectory

$$\psi = \pm\pi/2 + \theta \tag{4.37}$$

where the sign depends on whether the agent is in a clockwise or counterclockwise orbit, respectively. Thus,

$$\cos\psi = \cos(\pm\pi/2 + \theta) = \mp\sin\theta \tag{4.38}$$

and

$$\sin\psi = \sin(\pm\pi/2 + \theta) = \pm\cos\theta. \tag{4.39}$$

Finally, by Lemma 1 and Lemma 2 we have

$$\begin{aligned} \operatorname{atan2}\left(\begin{bmatrix}\cos\theta \\ \sin\theta\end{bmatrix} + -\frac{p_i}{\|p_i\|}\right) &= \arctan(\tan(\pm\pi/4+\theta)) && (4.40)\\ &= \pm\pi/4+\theta && (4.41)\end{aligned}$$

Therefore the angular velocity of an agent in an attraction-orientation torus is

$$\begin{aligned} \omega &= k(\pm\pi/4+\theta-\theta) && (4.42)\\ &= \pm k\pi/4. && (4.43)\end{aligned}$$

which is one-half the angular velocity of an attraction-only torus. □

Corollary 1. *The radius of an attraction-orientation torus with agent speed s and angular velocity gain k is $4s/k\pi$, which is twice the radius of an attraction-only torus.*

Proof. Given the constant angular velocity ω the time for an agent to complete one full revolution is $2\pi/\omega$. The distance traveled is $\frac{2\pi}{\omega}\cdot s$ which is equal to the circumference of the agent's orbit. Thus,

$$\frac{2\pi s}{\omega} = 2\pi r \tag{4.44}$$

and the radius of the torus is

$$\begin{aligned} r &= \frac{s}{\omega} && (4.45)\\ &= \frac{4s}{k\pi}. && (4.46)\end{aligned}$$

Thus, because the angular velocity of an attraction-only torus is twice as large as the angular velocity of an attraction-orientation torus, the radius of an attraction-orientation torus is twice the length of an attraction-only torus. □

4.4 Analysis of the flock attractor

The previous section showed that attraction dynamics along with a stationary centroid result in a stable cyclic behavior. Based on these results we propose that the torus group type is fundamentally caused by the attraction dynamics in the model and that the orientation and repulsion dynamics simply cause agents in a torus to rotate in the same direction and avoid collisions. In this section we propose that the flock group type is fundamentally caused by the orientation dynamics in our model, with attraction and repulsion simply causing the flock to stay cohesive but avoid collisions.

Consider a system with only orientation dynamics ($u_i = u_i^o$). The angular velocity $\dot{\theta}$ becomes

$$\dot{\theta}_i = k\left(atan2\left(v_i^y + \sum_j v_j^y, v_i^x + \sum_j v_j^x\right) - \theta_i\right). \tag{4.47}$$

We can approximate the right-hand side of Equation (4.47) using the following form of angular averaging

$$\begin{aligned} \dot{\theta}_i &= k\left(\frac{\theta_i + \sum_j \theta_j}{n_i + 1} - \theta_i\right) \\ &= k\frac{(\theta_i + \sum_j \theta_j) - (n_i + 1)\theta_i}{n_i + 1} \\ &= \frac{k}{n_i + 1}\sum_j (\theta_j - \theta_i) \end{aligned} \tag{4.48}$$

where n_i is the number of neighbors of agent i. When the underlying orientation graph is connected, (4.48) is known to cause all agents to converge to a common heading (see Chapter 2 in [34]). Thus, agents following the dynamics in Equation (4.48) will converge to a stable flock.

It may seem that the flock is simply a torus with an extremely large radius. However, the model dynamics contain no explicit reason for the flock to turn in one direction more often than the other. To verify this empirically, we ran 10 simulations for 1,000,000 time steps where the initial conditions were set to form a flock by setting all initial headings to

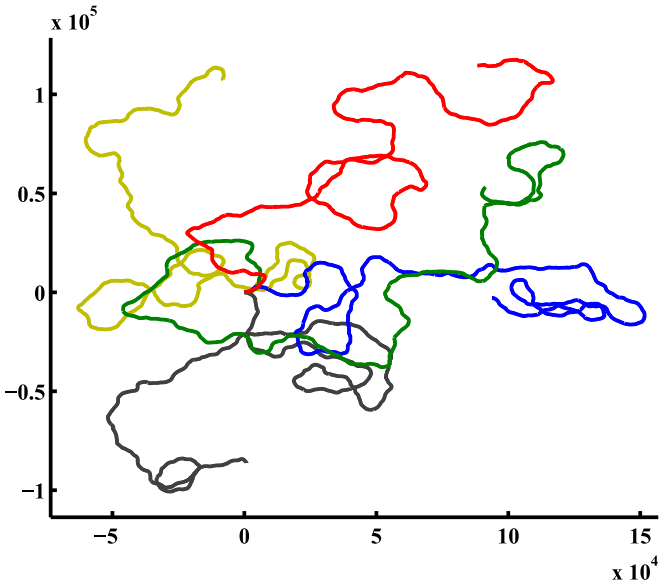


Figure 4.5: The trajectories of five simulations run for 100,000 seconds each with initial conditions set to form a flock starting close to the origin.

$\theta = 0$ with randomly placed initial positions within a 10×10 area centered at the origin. We plotted the trajectories of the centroids of several simulations to see whether there is any kind of pattern in the movement of a flock that appears torus-like. The trajectories of five of the simulations are shown in Figure 4.5.

These trajectories show that the movement of the flock is surprisingly random, and there does not appear to be any evidence that the flocks move in large circles. To confirm this observation we picked a single simulation and computed the distribution over the change in heading for the group. This was done using consecutive group centroid measurements to compute the group turning direction, $\Delta\theta_g(t)$, as follows

$$\Delta\theta_g(t) = \text{atan2}(c_g(t + \Delta t) - c_g(t)) - \text{atan2}(c_g(t) - c_g(t - \Delta t)) \tag{4.49}$$

where c_g is the group centroid. Figure 4.6 shows the distribution over $\Delta\theta_g$ for one of the simulations. Other simulations were almost identical. There appears to be no preference for the group to turn either right or left and the data appears to be normally distributed with

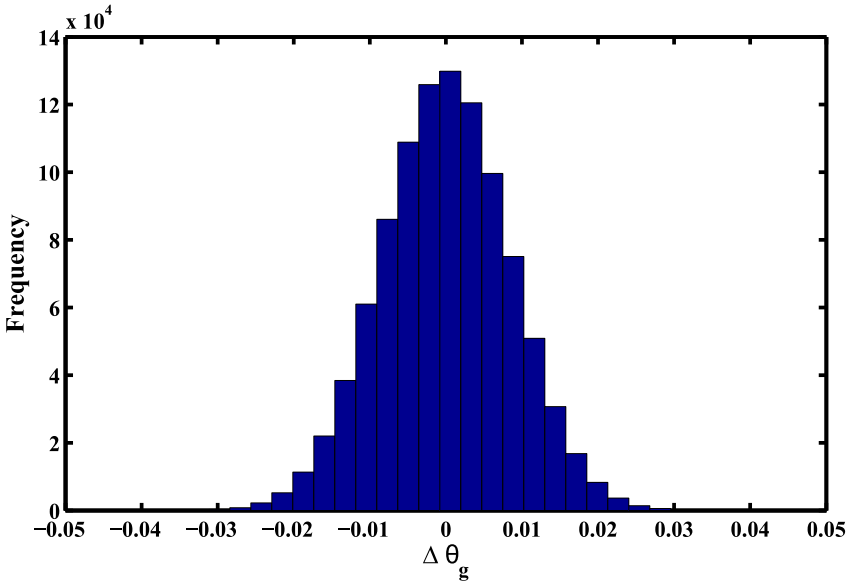


Figure 4.6: The distribution of $\Delta\theta_g$ over a single run for 100,000 seconds each with initial conditions set to form a flock. $N = 100$, $R_o = 8$.

zero mean. Therefore, based on our mathematical and empirical analysis, we conclude that the flock is a second distinct attractor of our swarm model.

4.5 Summary

We have shown that the swarm model presented in Section 3.1 exhibits hysteresis which makes it difficult to change the collective behavior of the swarm through parameter changes. We then demonstrated a set of parameters for the model that allow both a flock and a torus to form with equal probability to facilitate changing the collective behavior of the group with out requiring global parameter broadcasts or inter-agent communication. Finally, we presented results that support our claim that the model has two fundamental attractors. We claim that a torus is fundamentally caused by attraction dynamics, and that a flock is fundamentally caused by orientation dynamics. Because our model uses both attraction and orientation, both group types can emerge. Which group type emerges depends on the initial conditions of the model as well as the radius of orientation which determines the relative strength of attraction or orientation on the individual agents.

Chapter 5

Measuring Human-Swarm Interactions

An important part of this thesis will involve measuring and quantifying human-swarm interactions. Several researchers have proposed metrics for measuring human-robot and human-swarm interactions. Olsen and Goodrich propose two metrics of human-robot interaction: *interaction time* and *neglect time* [35]. Interaction time is defined as the time it takes for a human to bring a robot's performance to some desired level. Neglect time is defined as the time it takes for the robot's performance to fall below some threshold when left unattended. We will focus mainly on applying the notion of interaction effort to human-swarm interactions, but note that other work [36] also investigates the notion of neglect time when applied to human-swarm interactions.

5.1 Interaction effort

Kerman proposes a metric for measuring human-swarm interaction called *interaction effort* [36]. Interaction effort measures the number and duration of interactions required to bring a robot swarm's performance to some desired level. It is defined as

$$IE = M \cdot IT \tag{5.1}$$

where we use M as the number of agents receiving human influence and IT as the interaction time. In this thesis we specifically define interaction time as the time it takes to reduce the performance error of the swarm below some desired threshold. When controlling a robot swarm, we measure the performance error as the swarm's collective deviation from some

desired behavior. For example, this could be the difference between the actual and desired group type, group centroid, or group heading.

5.2 Observation effort

When observing samples from individual agents we want to measure the "effort" required to determine the collective state of a swarm. We propose *observation effort* as a measure of the bandwidth and time required to estimate a global or collective property of a swarm. In this thesis we first focus on classifying the collective group type of a swarm. For classification tasks, we define observation effort as the effort required to obtain a classification accuracy above a certain threshold. Observation effort is calculated in terms of the number of agents sampled and the length of time they are sampled and is measured in units of $agents \cdot seconds$. Mathematically, we define observation effort as

$$OE = Z \cdot OT \tag{5.2}$$

where Z is the number of observed agents and OT is the observation time, the time required to obtain a certain classification accuracy.

We also investigate using local samples from individual agents to estimate global properties such as the group heading and group centroid. In this case we do not have an easy way of measuring accuracy and instead measure the error of our estimate. Thus, we define a metric similar to observation effort and call it *estimation effort*. Estimation effort (EE) is defined as

$$EE = Z \cdot ET \tag{5.3}$$

where Z is the number of observed agents and ET is the estimation time, the time required to reduce the estimation error below some threshold.

Chapter 6

Characterization of Collective Behavior

In this chapter we examine several global indicators of the collective behavior of a swarm and show how to classify these behaviors using local information from a small number of agents. We present a naive Bayes classifier for detecting stable group types that achieves high accuracy while requiring low observation effort. Finally, we present evidence for *neglect benevolence* [37], the benefit of waiting for collective behaviors to stabilize before interacting with the swarm, when classifying collective behaviors. Estimating the centroid and heading of a swarm is investigated later in Chapter 7.

6.1 Global indicators of collective behavior

In Section 3.2 we introduced two different global measurements of collective behavior that can be used to determine the group type (flock or torus) of a swarm: group polarization and group momentum. However, neither of these measurements take into account the graph structure of the agent interactions. We propose using the Fiedler eigenvalue of the underlying graph topology formed by agent interactions as another way of measuring differences in a swarm's collective behavior. The underlying graph topology in multi-agent systems is known to be important for the convergence of consensus protocols and the speed of consensus is directly related to the magnitude of the Fiedler eigenvalue [4]. The Fiedler eigenvalue, ν_{n-1}, is defined as the second smallest eigenvalue of the graph Laplacian and is a measure of the connectedness of the graph [38]. The Laplacian L of an undirected graph is defined as

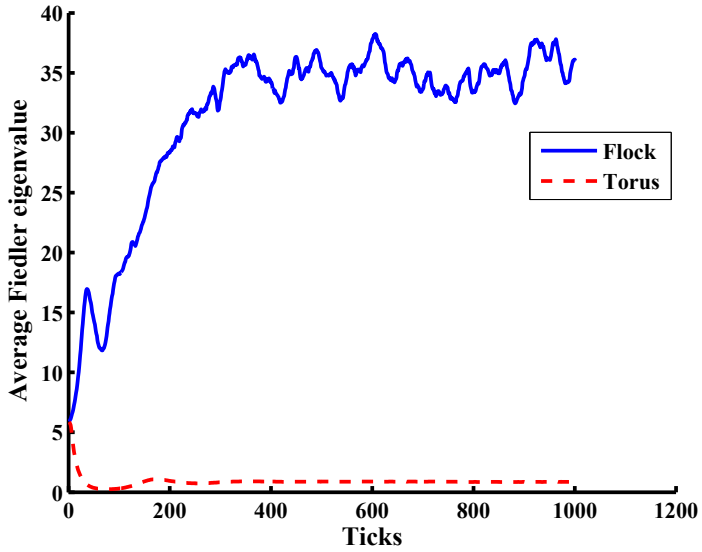


Figure 6.1: The average Fiedler eigenvalue over 100 simulations of a flock and a torus. Results show that the graph structure of the two group types are markedly different.

$L = D - A$ where A is the adjacency matrix induced by the agent interactions and D is the diagonal matrix with agent degrees along the diagonal.

Figure 6.1 shows the average Fiedler eigenvalues per simulation tick for flock and torus group types started from random initial conditions. Because the agent interactions are stochastic, the underlying graph is directed. To create an undirected graph representation of the swarm we used the radius of orientation $R_o = 8$ to determine the neighborhood of each agent where $A_{ij} = 1$ if and only if $\|p_i - p_j\| < R_o$. The figure shows that there are some transient dynamics for the first 300 ticks while the group types stabilize. After stabilizing, the torus Fiedler eigenvalue is relatively constant, whereas the flock Fiedler eigenvalue is much larger and has more variation. These results indicate that the Fiedler eigenvalue is a good measure of the topological difference between the flock and the torus attractors of our swarm model.

6.2 Local classification

Our model produces two stable group types: the flock group type and the torus group type. We have discussed three global features that distinguish between these group types: group polarization, group momentum, and the Fiedler eigenvalue. These metrics require collecting information about every agent in the swarm and do not appear useful in the context of scalable human-swarm interactions, where we assume we can only interact with a small percentage of the agents. However, there are related local measures that can be used instead. The group polarization and momentum are related to the individual agent headings and angular velocities, respectively. The Fiedler eigenvalue, ν_{n-1}, of a graph is related to the degree sequence $(d_1, d_2, \ldots, d_n)$ of the graph by the following relationship [38]

$$0 \leq \nu_{n-1} \leq \frac{n}{n-1}\delta \leq \frac{n}{n-1}\bar{d} \tag{6.1}$$

where $\delta = \min(d_1, d_2, \ldots, d_n)$, $\bar{d} = \frac{1}{n}\sum_i^n d_i$, and the degree of an agent is given by the number of neighbors it interacts with. This relationship shows that ν_{n-1} is bounded above by the minimum and average degree, which can both be estimated by sampling individual agent degrees.

We propose that a small sample of agent angular velocities and numbers of neighbors (degrees) provides sufficient information to detect the group type of the swarm. Angular velocity from a few agents is used to determine whether the agents are turning at approximately the same rate (torus) or moving straight (flock). The degree, or number of neighbors, of a few agents indicates whether the swarm's graph structure has high (flock) or low (torus) connectivity. We have chosen to focus on these two features because neither require any kind of localization or shared frame of reference—they only require that an agent knows how fast it is turning and how many other agents are nearby. We assume that the angular velocity is estimated using a single-axis gyroscope, or using dead reckoning. We further assume that the

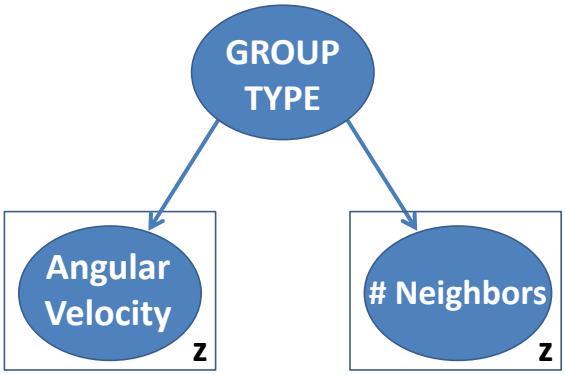


Figure 6.2: Bayesian network for group type classification using samples of local information from Z agents.

number of neighbors is estimated using some kind of computer vision system. We leave the details as future work.

6.3 Bayesian classification of swarm group type

We make the Naive Bayes assumption that all features are conditionally independent given the group type resulting in the graphical model shown in Figure 6.2. Using the two local features discussed in Section 6.2, number of neighbors and angular velocity, we compute the Bayesian optimal decision of the actual group type, $type_g$ as follows

$$type_g = \underset{type}{\operatorname{argmax}}\, P(type) \prod_{t_s=1}^{OT} \prod_{i=1}^{Z} P\big(d_i(t_s) \,\big|\, type\big) P\big(\omega_i(t_s) \,\big|\, type\big) \tag{6.2}$$

where $type \in \{clockwisetorus, counterclockwisetorus, flock\}$, OT is the observation time, Z is the number of agents sampled, and $d_i(t_s)$ and $\omega_i(t_s)$ are the degree and angular velocity of agent i sampled at time t_s, respectively. Equation (6.2) selects the group type that has maximum probability given the prior probabilities of the group types and the observed local features in Figure 6.2, namely, agent degrees and angular velocities. The individual likelihoods $P\big(d \,\big|\, type\big)$ and $P\big(\omega \,\big|\, type\big)$ are learned from training data.

6.4 Group type classification data sets

To evaluate the accuracy of group type classification we created a training and a test set, both consisting of simulations for the flock and torus group types. We then evaluated the accuracy of several different classifiers. All simulations used the parameters discussed in Section 4.2.

6.4.1 Training data

To obtain data to model the numbers of neighbors and angular velocities of the flock and torus group types we ran three experiments: one for the flock, one for the clockwise torus, and one for the counterclockwise torus. Each experiment consisted of 100 replicates of 100 seconds each. For the flock simulations each agent's initial heading was set to $\theta = 0$ and the agents' starting locations were randomly chosen in the interval $[-10, 10] \times [-10, 10]$. These parameters were used because we found that they ensured that the swarm formed a flock. The initial positions for the torus simulations were also chosen randomly, but the initial headings were chosen such that

$$\theta_i = \text{atan2}(y_i, x_i) \pm \frac{\pi}{2} \tag{6.3}$$

where the sign of $\pi/2$ was chosen depending on the desired orientation of the torus. This was done to ensure that the swarm formed a torus. For each simulation we let the group type stabilize for 25 seconds and then recorded the number of neighbors and angular velocity for each agent in the group.

6.4.2 Test data

We created a test set by running 100 replicates with random initial headings and positions so that each simulation could produce either a flock or a torus. Out of the 100 simulations, 53 formed a torus and 47 formed a flock where the convergence was checked using the final

	Mean	Standard Deviation
Counterclockwise Torus	10.1172	2.8638
Clockwise Torus	10.1061	2.8592
Flock	28.9906	7.7746

Table 6.1: The sample mean and standard deviation for the distributions of number of neighbors over natural group types.

m_{group} and p_{group} for each replicate. A simulation was labeled as a torus if $m_{group} > 0.75$ and $p_{group} < 0.25$. A simulation was labeled as a flock if $m_{group} < 0.25$ and $p_{group} > 0.75$.

6.5 Description of classifiers

In the following sections we examine several different classifiers and compare the resulting classification accuracy and observation effort of each classifier. We first look at the problem of classifying the swarm's group type as either a flock or a torus, ignoring the rotation of the torus. We investigate the classification accuracy and observation effort when only using the number of neighbors as a feature, when only using absolute angular velocity as a feature, and when using both features together. We then look at the problem of classifying the swarm's group type as either a flock, a clockwise torus, or a counterclockwise torus. We investigate the classification accuracy and observation effort when only using angular velocity as a feature and when using both the number of neighbors and angular velocity as features.

6.6 Classification using number of neighbors

We first consider the problem of group type classification using only the number of neighbors as a feature. Using our training data we computed sample means and sample standard deviations of the different group types as shown in Table 6.1. As expected, we see that because of the symmetry in the model, the statistics for counterclockwise and clockwise torus group types are essentially identical. Because of this symmetry, we arbitrarily chose to use the statistics for the counterclockwise torus as typifying the statistics for a general

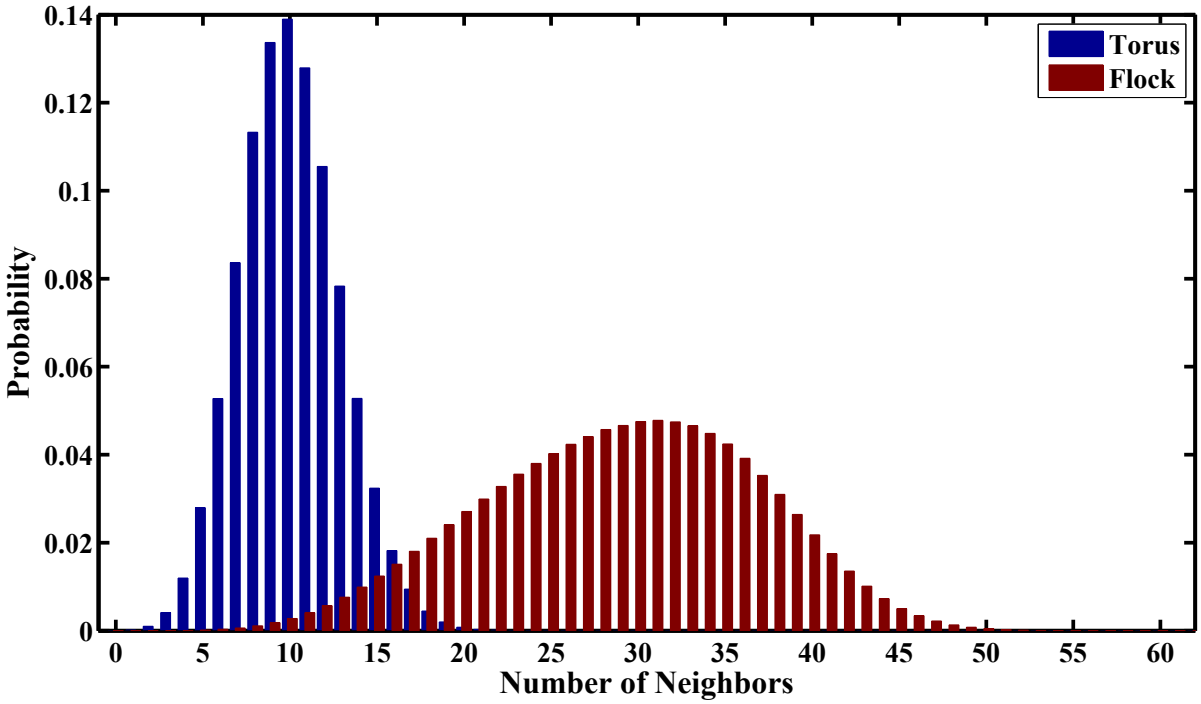


Figure 6.3: Distributions over an agent's number of neighbors in torus and a flock. $N = 100$ agents.

torus formation, regardless of the orientation. The actual sampled numbers of neighbors in the training data are plotted as histograms in Figure 6.3 where the frequencies have been normalized to obtain probabilities. This figure shows visually that each distribution has a unique mean and standard deviation and that that there is some overlap between the distributions, mainly between 10 and 20 neighbors. Because sampling the number of neighbors of agents in the swarm gives us no information about the orientation of the torus, we will only be able to perform a binary classification to detect whether the swarm is in the flock or torus formation.

6.6.1 Naive Bayes formulation

To perform classification using only samples of number of neighbors, we used the following reduced version of Equation (6.2) to determine the group type

$$type_g = \underset{type \in \{torus, flock\}}{\operatorname{argmax}} P(type) \prod_{t_s=1}^{OT} \prod_{i=1}^{Z} P\big(d_i(t_s) \,\big|\, type\big) \tag{6.4}$$

$$= \underset{type \in \{torus, flock\}}{\operatorname{argmax}} \sum_{t_s=1}^{OT} \sum_{i=1}^{Z} \log P\big(d_i(t_s) \,\big|\, type\big) \tag{6.5}$$

where $d_i(t_s)$ is the degree or number of neighbors of agent i at sample time t_s, OT is the observation time, and Z is the number of distinct agents sampled. The second line results from using log probabilities and from the fact that our test data was produced using simulation parameters that have an equal chance of producing a flock or torus. Thus, we assume $P(torus) = P(flock) = 1/2$. Logarithms are used to avoid loss of precision.

To use Equation (6.5) we need to estimate the likelihood $P\big(d \,\big|\, type\big)$ of an agent having a certain number of neighbors d, given that the group is in a torus or a flock formation. To do this we use a simple maximum likelihood estimate using normalized counts from the training data. This results in the estimate

$$P(d \mid type) = \frac{Count(d, type)}{\sum_{j=0}^{N-1} Count(j, type)}. \tag{6.6}$$

where d is the number of neighbors and $type \in \{torus, flock\}$.

6.6.2 Classification accuracy and observation effort

Because the test data is generated from simulations that started at random initial conditions, there can be a large amount of transient behavior before the group type stabilizes. To give ample time for a group type to form we gave the simulations 90 seconds to form and used the last 10 seconds (100 time steps) for computing the classification accuracy and observation

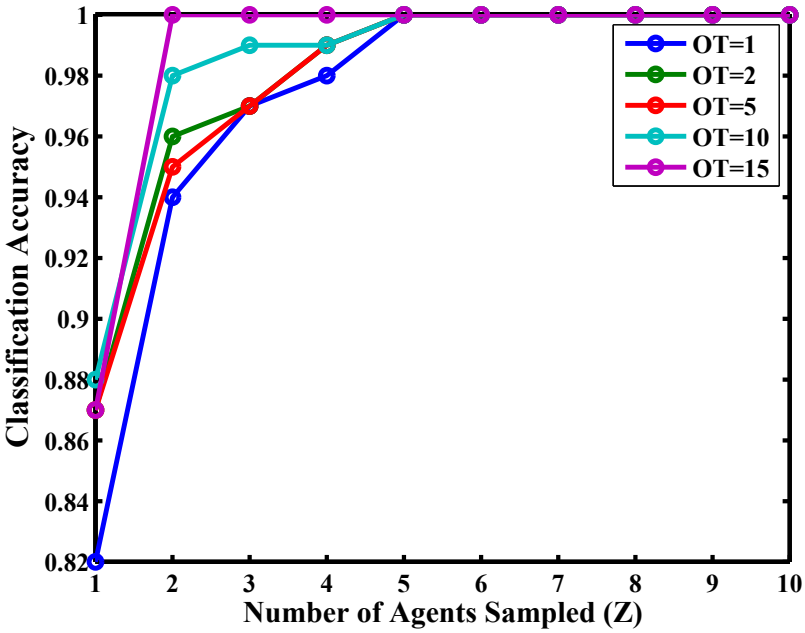


Figure 6.4: Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying whether the group type is a torus or flock using the number of neighbors as the only feature. N = 100 agents.

effort. We experimented with values of Z ranging from 1 to 10 and OT ranging from 1 to 100. Figures 6.4 and 6.5 show the classification accuracy over different values of Z and OT.

Figure 6.4 shows that, as Z increases, the accuracy quickly reaches 100% on the test set. Furthermore increasing the observation time OT for a particular value of Z tends to also increase the accuracy. It is surprising that, once 5 or more agents are sampled, the detection algorithm correctly classified every group type in the test set regardless of the length of observation time. Also interesting to note is that when sampling for 15 time steps, data from only 2 agents was sufficient to achieve 100% accuracy on the test set.

Figure 6.5 shows the classification accuracy as a function of observation time for Z between 1 and 5. Interestingly, the results show that for $Z = 1$ increasing the observation time is not beneficial past an observation time of 5 and that the accuracy varies between 87 and 88% for $Z = 1$ and $OT \geq 5$. The reason for this is explored further in Section

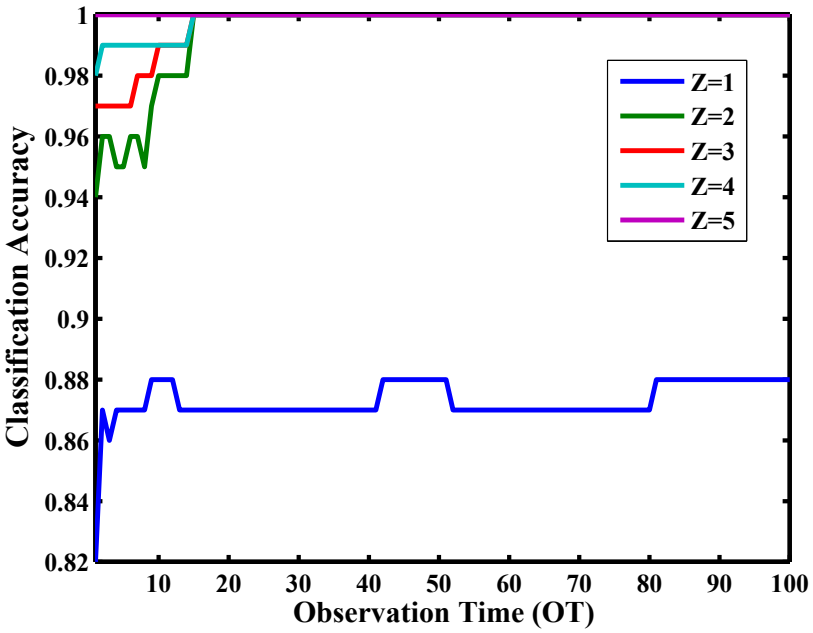


Figure 6.5: Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying whether the group type is a torus or flock using the number of neighbors as the only feature. N = 100 agents.

6.6.3. The variations in accuracy occur because the number of neighbors is stochastic so the classifier may switch back and forth between classifying the swarm as a torus or as a flock as new samples are added. For $Z > 1$ we see that increasing observation time resulted in 100% accuracy for $OT \geq 15$. Additionally sampling 5 or more agents resulted in perfect classification accuracy even for $OT = 1$. These results show that even for a severely limited bandwidth of $Z = 2$, very high accuracy can be achieved if time constraints allow sampling the agents several times. These results also show that if more bandwidth is available, then high accuracy can be achieved even faster by simply taking a small number of samples from multiple agents.

To compute the observation effort for group type detection using the number of neighbors, we computed the observation time needed to first reach a certain accuracy μ for a given number of agents sampled. If the accuracy never reached the threshold μ then the observation effort was deemed infinite. The results of this analysis are shown in Figure 6.6.

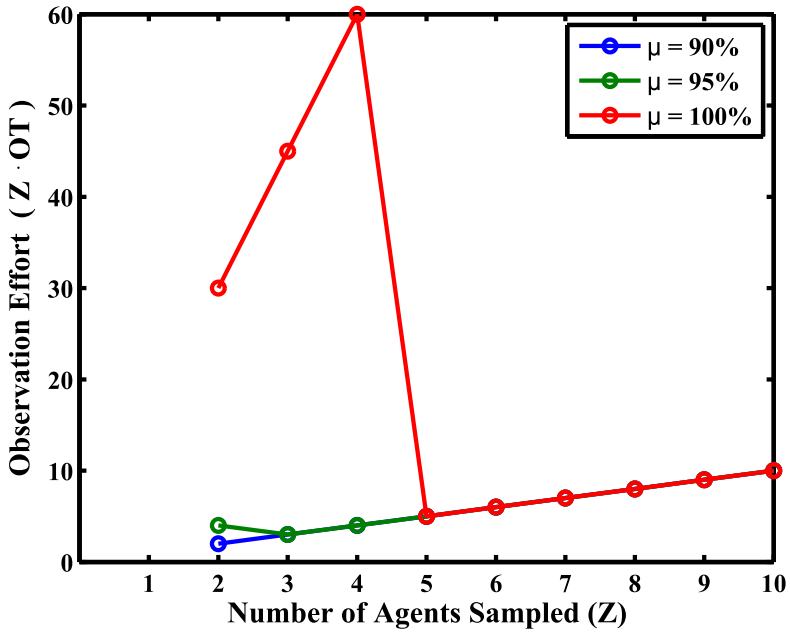


Figure 6.6: Observation effort for group type classification over different accuracy thresholds μ when using the number of neighbors sampled from Z agents. Using $Z = 1$ never resulted in the desired accuracy so the observation effort was infinite and is not shown. Results are for N = 100 agents.

Analyzing the observation effort shown in Figure 6.6 we see that the observation effort is strictly increasing and identical for $Z \geq 5$ for $\mu = 90$, 95, and 100%. The larger observation efforts for $Z = 2$, 3, and 4 are because the accuracy did not reach 100% on the test set until $OT = 15$. The sharp drop off from $Z = 4$ to 5 is because for $Z \geq 5$, 100% accuracy was achieved using only one sample from each agent. Table 6.2 summarizes the values of Z that minimized observation effort for the different thresholds μ.

	$\mu = 90\%$	$\mu = 95\%$	$\mu = 100\%$
Minimum OE	2	3	5
Corresponding Z	2	3	5

Table 6.2: Values of Z that minimized observation effort for $\mu = 90$, 95, and 100% accuracy on the test set using only the number of neighbors as a feature.

We see that there is an exact correspondence between OE and Z. It makes sense that there is a linear relationship between OE and Z because samples from a single agent will

typically be very similar over a short period of time. On the other hand, sampling from more agents will result in local samples from different areas of the swarm, so increasing Z rather than OT results in a more uniform sampling of the swarm resulting in better classification accuracy.

6.6.3 Error analysis

We performed a detailed error analysis on a few of the results for group type classification using only the number of neighbors. We summarize some of the main results here and refer the reader to Appendix B.1 for more details.

We found that using only the number of neighbors as a feature results in a classification bias toward misclassifying a flock as a torus when $Z = 1$ or 2. Examining the simulations from the test set that were incorrectly classified, we found that the main reason that a flock is often misclassified as a torus turns out to be the long thin tail that often forms behind a flock when agents start from random initial conditions. As shown in Figure 6.7, when sampled agents are in the tail they have fewer neighbors and our classifier selects torus as the most likely group type. It should be noted, however, that given enough time the flock will fully clump together eliminating the long tail and forming a flock similar to that shown in Figure 2.1(c). We also found several instances where misclassifications occurred because of the inherent noise caused by our stochastic topology—agents occasionally have a much larger (or smaller) number of neighbors than is typical for a certain group type.

Based on these common common errors when sampling only a few agents, we see that adding in additional information about the agent's angular velocity would appear to improve the probability of detection. Additionally, as shown by the results in figures 6.4 and 6.5, we see that increasing the number of agents sampled provides a large increase in accuracy, but because agents in a long tail will take time to catch up to the group, simply sampling one agent multiple times does not have a significant impact on accuracy. These results also show that our classification method based on an agent's number of neighbors is robust to

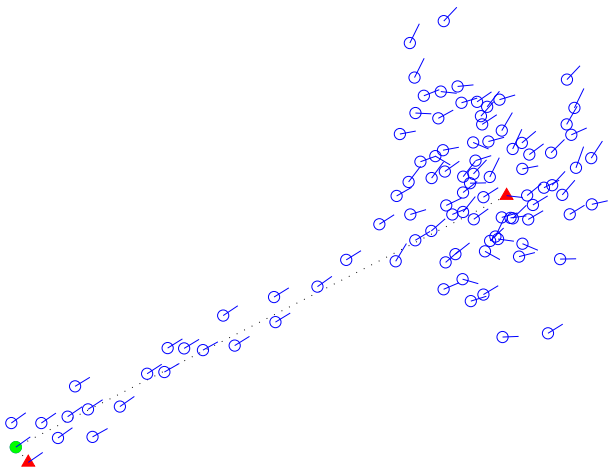


Figure 6.7: An agent sampled (the green disc) from the long tail of flock has fewer neighbors (the 2 red triangles) than those in the main clump of the flock resulting in the misclassification of the flock as a torus.

	Mean	Standard Deviation
Counterclockwise Torus	0.3829	0.2001
Clockwise Torus	-0.3829	0.1996
Flock	-0.0004	0.6869

Table 6.3: The sample mean and standard deviation for the distributions of angular velocity shown in Figure 6.8

the inherent noise caused by our stochastic topology as long as we can sample from multiple agents.

6.7 Classification using angular velocity

The angular velocities from the training data are plotted as histograms in Figure 6.8. The cut-off for angular velocity, as mentioned in Section 3.1, is $k\pi$ which in our case works out to be $\pi/2 \approx 1.5708$. This is the reason for the sharp cut-offs shown in Figure 6.8. The sample means and sample standard deviations of each group type in the training data are reported in Table 6.3.

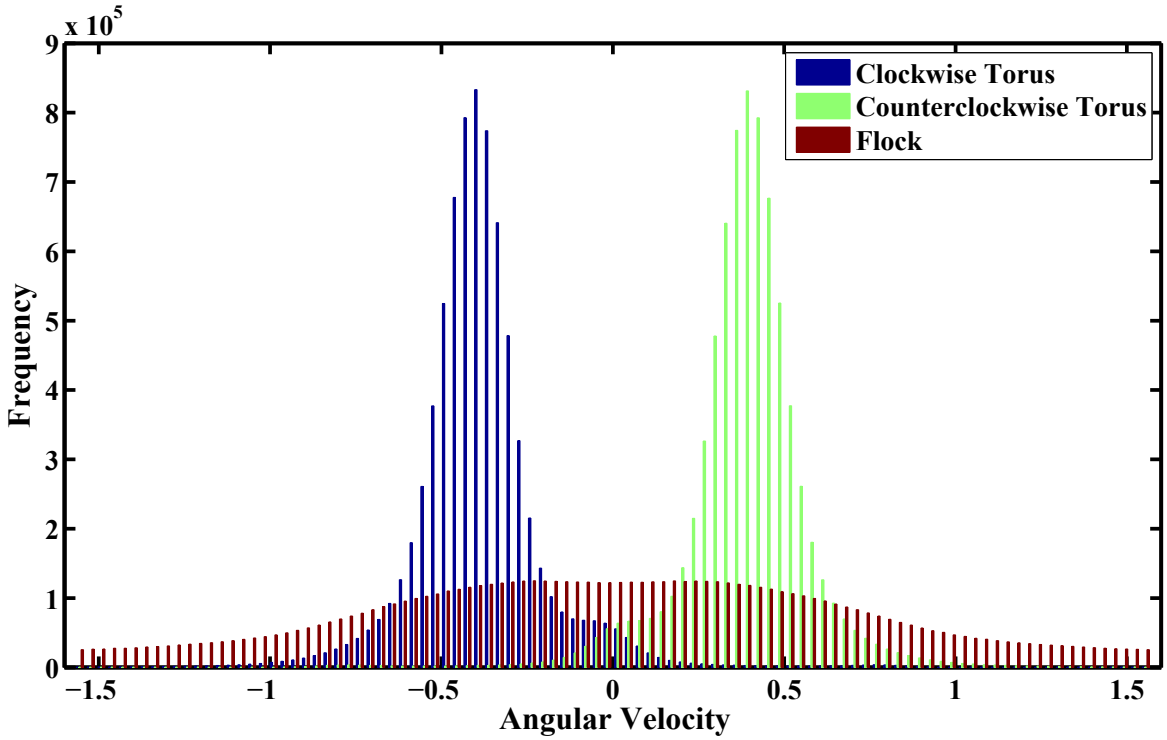


Figure 6.8: Distributions over an agent's angular velocity (radians/second) in a clockwise torus, counterclockwise torus, and a flock. Simulation parameters were $N = 100$, $R_o = 8$, $R_r = 1$, $s = 5$, $k = 0.5$.

The statistics for the angular velocity of a flock are very intuitive; the average is almost zero and the there is a large deviation caused by agents moving around in the swarm. The statistics for the counterclockwise and clockwise torus are less intuitive, but make sense after doing some analysis. As shown in Section 4.3.2, when orientation is added to attraction the angular velocity is $\pm k\pi/4 \approx \pm 0.3927$. Accounting for the effects of the random topology and repulsion we see that a sample mean of 0.3829 is very close to our theoretical result and provides a good sanity check for the results we have obtained. Because of the symmetry in the model we also see that the distributions for the clockwise and counterclockwise torus formations are simply mirror images of each other.

When using samples of angular velocity we have two choices, we can take the absolute angular velocity and perform binary classification between a flock and a torus or we can use

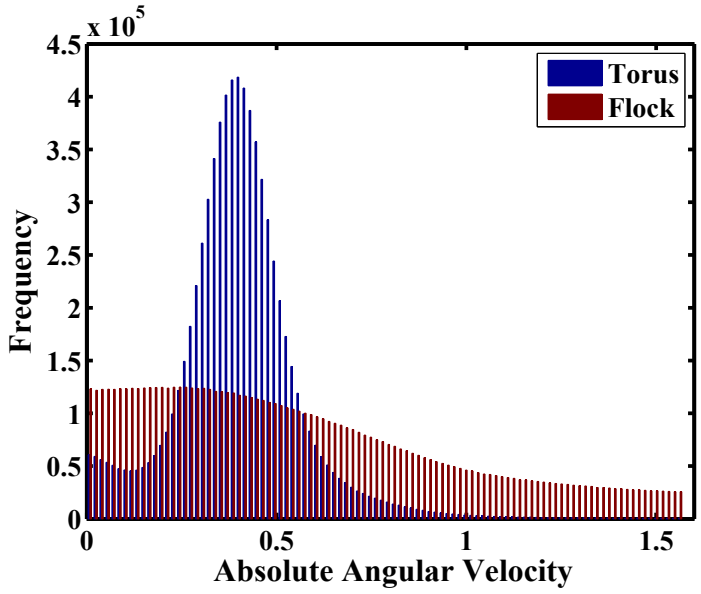


Figure 6.9: Distributions over an agent's absolute angular velocity (radians/second) in a torus and a flock. Simulation parameters were $N = 100$, $R_o = 8$, $R_r = 1$, $s = 5$, $k = 0.5$.

	Sample Mean	Sample Standard Deviation
Torus	0.3996	0.1644
Flock	0.5608	0.3967

Table 6.4: The sample mean and variance of the absolute angular velocity of agents in a flock and torus.

signed angular velocities to actually detect the rotation of the torus. We first consider using only absolute angular velocity as a feature.

6.7.1 Absolute angular velocity

Figure 6.9 shows a histogram of the absolute angular velocities for the flock and torus formations. Table 6.4 shows the sample mean and standard deviations. We used the following classification scheme

$$type_g = \underset{type \in \{torus, flock\}}{\operatorname{argmax}} \sum_{t_s=1}^{OT} \sum_{i=1}^{Z} \log P\big(|\omega_i(t_s)| \,\big|\, type\big) \tag{6.7}$$

where $\omega_i(t_s)$ is angular velocity of agent i at sampling time t_s. To obtain an estimate of $P\big(|\omega| \,\big|\, type\big)$, we first tried using normal distributions to model the angular velocity, but found that simply discretizing the angular velocity and using normalized counts achieved better accuracy. For this reason we only report the latter results. Additionally, discretizing the angular velocity distributions allows us to easily combine angular velocity (a continuous feature) with number of neighbors (a discrete feature). To discretize angular velocity we binned the data every 0.1 radians from 0 to 1.6 creating a discretization of 16 bins. This corresponds to bin widths of 5.7296 degrees/second. We took each sample of angular velocity for each group type from the training data, took the absolute value, and then created a discrete probability distribution by binning all of the training data and normalizing the counts to compute probabilities.

6.7.2 Classification accuracy and observation effort

The resulting classification accuracies using discretized absolute angular velocity are shown in figures 6.10 and 6.11. Figure 6.10 shows that there is a definite increase in accuracy as Z and OT increase but that the classification accuracy is lower than for number of neighbors. Figure 6.11 shows that increasing observation time when sampling absolute angular velocity increases classification accuracy more than when sampling the number of neighbors, but that it does not reach 100% very quickly. Interestingly for $Z = 1$, increasing the observation time raises the accuracy from 69% to 98%, whereas for number of neighbors the accuracy for $Z = 1$ started at 82% accuracy but only increased to 88% when observation time was increased. Our previous error analysis showed that this occurred because occasionally the single agent sampled from the swarm was in the long thin tail of the flock as shown in Figure 6.7. Because the simulations were only run for 100 seconds this did not allow enough time for the tail to catch up to the main body of the flock and resulted in low samples of numbers of neighbors for the entire simulation. On the other hand, agents in the tail of a flock move in a relatively

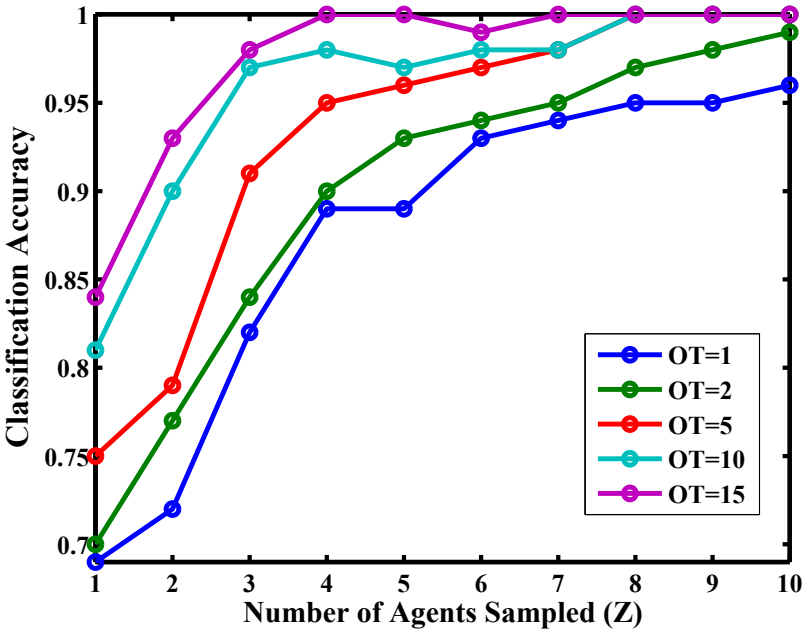


Figure 6.10: Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying whether the group type is a torus or flock using absolute angular velocity as the only feature. N = 100 agents.

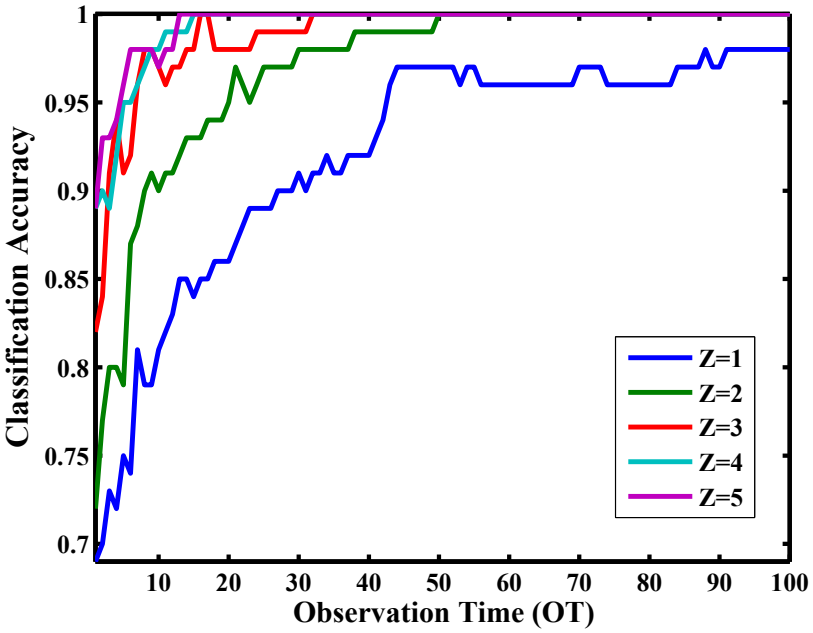


Figure 6.11: Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying whether the group type is a torus or flock using absolute angular velocity as the only feature. N = 100 agents.

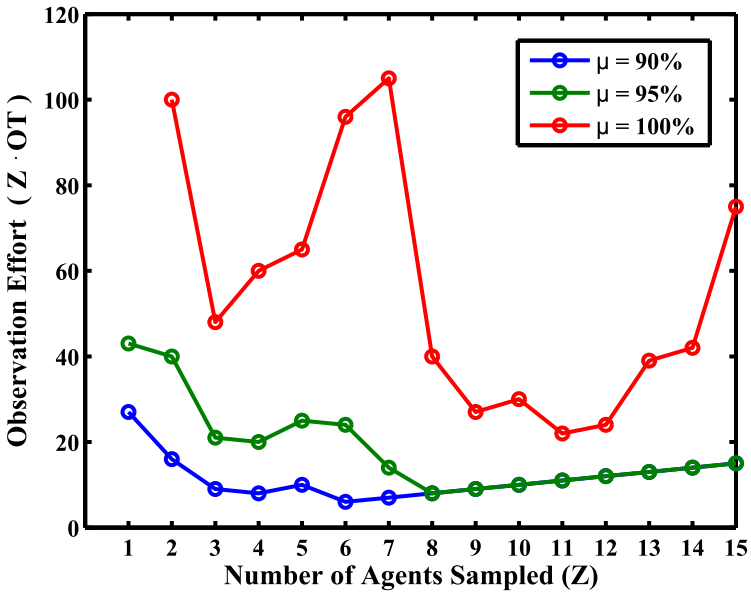


Figure 6.12: Observation effort for group type classification for different accuracy thresholds when sampling absolute angular velocity from Z. Results are for N = 100 agents.

	$\mu = 90\%$	$\mu = 95\%$	$\mu = 100\%$
Minimum OE	6	8	22
Corresponding Z	8	3	11

Table 6.5: Values of Z that minimized observation effort (OE) for μ = 90, 95, and 100% accuracy on the test set using only absolute angular velocity as a feature.

straight line in order to catch up with the body of the flock. Thus, absolute angular velocity samples are close to zero and result in a higher classification accuracy over time when using absolute angular velocity as a feature with $Z = 1$.

Figure 6.12 shows the observation effort for group type classification using absolute angular velocity. Analyzing the observation effort shown in Figure 6.12 we see that there is much more variation in observation effort over the test set when sampling angular velocity than was found when sampling the number of neighbors. Additionally, we see that for $Z = 1$ the observation effort is finite for $\mu = 90\%$ and $\mu = 95\%$. Table 6.5 summarizes the values of Z that minimized observation effort for the different thresholds μ. Unlike classification when using number of neighbors, we see that there is not a strict linear relationship between OE

and Z. Additionally, we see that the observation effort required when using only absolute angular velocity is significantly higher than classification using only the number of neighbors. The reader is referred to Appendix B.2 for more details on the classification errors.

6.8 Combining number of neighbors and absolute angular velocity

The previous sections showed that both the number of neighbors and the absolute angular velocity can be used individually to classify a swarm's group type. We also showed that using only number of neighbors as a feature requires a smaller observation effort than using only absolute angular velocity. We investigated using both features in one classifier to see if we could achieve higher accuracy than using them individually. However, we found that the resulting classification accuracy and observation effort were nearly identical to those reported in Section 6.6.2 when using the number of neighbors as the only feature. Based on these results and the results from Sections 6.6 and 6.7, we conclude that using only the number of neighbors as a feature is sufficient for accurate group type classification with low observation effort if only torus or flock distinction is required.

6.9 Classifying group type and rotation

We next turn to the task of classifying the orientation of the torus group type. We define two rotations for the torus: counterclockwise and clockwise. Despite the high accuracy and low observation effort that results from using the number of neighbors to classify the group type, sampling numbers of neighbors gives us no information regarding the collective rotation of the swarm. Thus, sampling angular velocity becomes a critical feature if we want to classify group type rotation.

To determine the true orientation of a swarm we checked the group momentum and group polarization to determine whether it was a torus and then computed the sum of the sign of the final angular velocities. If that sum was less than zero we designated the group type as a clockwise torus and if the sum was positive we designated the group type as a

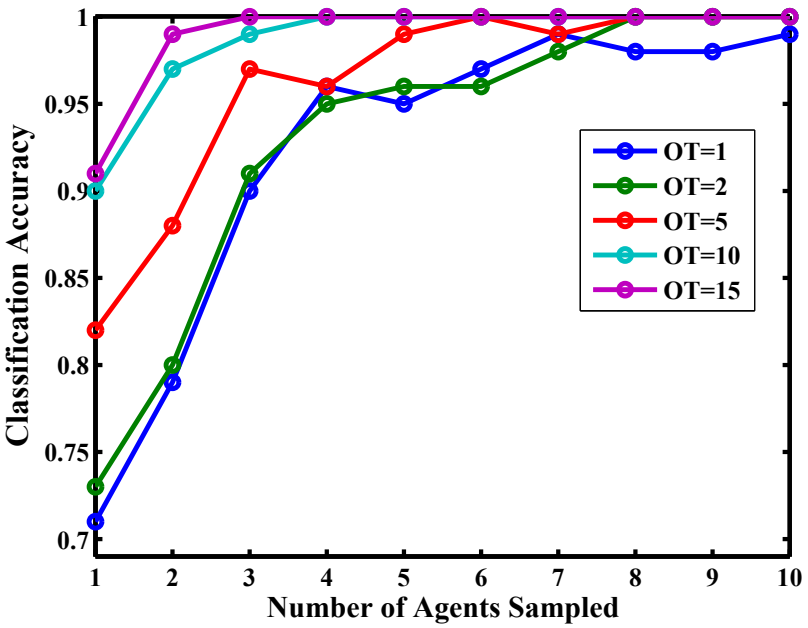


Figure 6.13: Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying the group type and rotation using angular velocity as the only feature. N = 100 agents.

counterclockwise torus. We examined our test set and found that out of the 53 simulations that formed a torus, 26 formed a clockwise torus and 27 formed a counterclockwise torus.

6.9.1 Classification using only angular velocity

We first investigated how well angular velocity does in classifying between a flock and the two possible rotations for a torus. Once again we discretized angular velocity values using bins widths of .1 radians from -1.6 radians to 1.6 radians resulting in 32 bins. The classification accuracies are shown in figures 6.13 and 6.14. Examining Figure 6.13 we see that classification accuracy increases quickly as Z increases and reaches 100% accuracy for $OT > 1$ for $Z \geq 8$. Figure 6.14 shows that increasing OT increases the classification accuracy for all values of Z. Additionally, we see that for all Z values shown, the accuracy reaches 100% as OT increases. Interestingly, the classification accuracy when using signed angular velocity to detect group

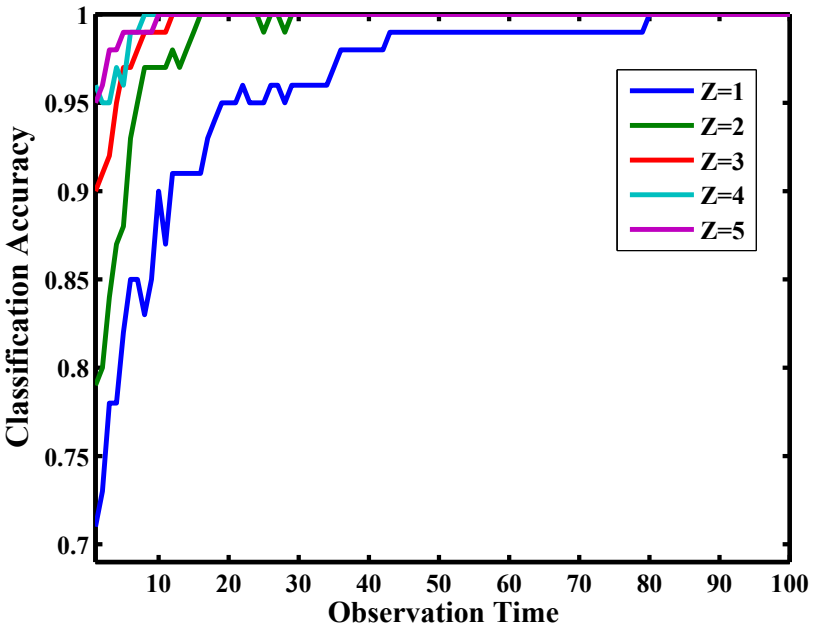


Figure 6.14: Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying the group type and rotation using angular velocity as the only feature. N = 100 agents.

	$\mu = 90\%$	$\mu = 95\%$	$\mu = 100\%$
Minimum OE	3	4	22
Corresponding Z	3	4	11

Table 6.6: Values of Z that minimized observation effort (OE) for μ = 90, 95, and 100% accuracy on the test set using only angular velocity as a feature.

type and group rotation is actually better than using absolute angular velocity to distinguish between a flock and a torus.

The observation effort is shown in Figure 6.15. We see that the observation effort is actually defined for $Z = 1$ for all thresholds μ which was not the case for either number of neighbors or absolute angular velocity. Table 6.6 summarizes the values of Z that minimized observation effort for the different thresholds μ. The reader is referred to Appendix B.3 for a more detailed error analysis.

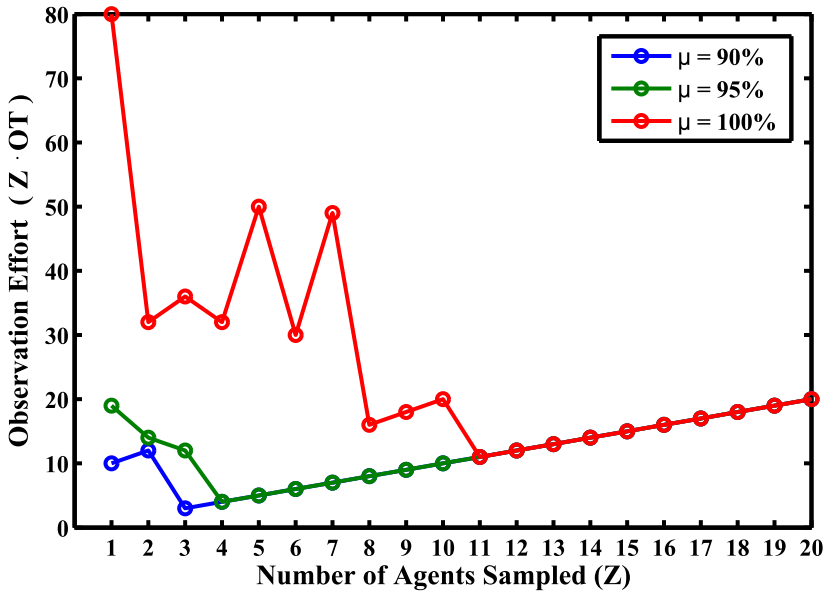


Figure 6.15: Observation effort when classifying group type and rotation using angular velocity sampled from Z agents and for different accuracy thresholds μ. Results are for N = 100 agents.

6.10 Combining angular velocity and number of neighbors to classify group type and rotation

For this final classifier we used both the number of neighbors and the discretized angular velocity as features in order to classify the group type and torus rotation. The classification accuracy results are shown in figures 6.16 and 6.17. These results show that adding number of neighbors as a feature dramatically increases the accuracy as number of agents sampled increases. It is interesting to note that, while combining angular velocity and the number of neighbors allows tri-group type detection with very high accuracy for small OT, Figure 6.17 shows that the classification accuracy never rises above 89% when $Z = 1$. Thus, we run into the same problem we had when doing binary group type classification: sampling a single agent in the tail of a flock lowers the classification accuracy. However, the results show that this classifier achieves a higher accuracy than the other classifiers we have considered, as long as we can sample from more than one agent.

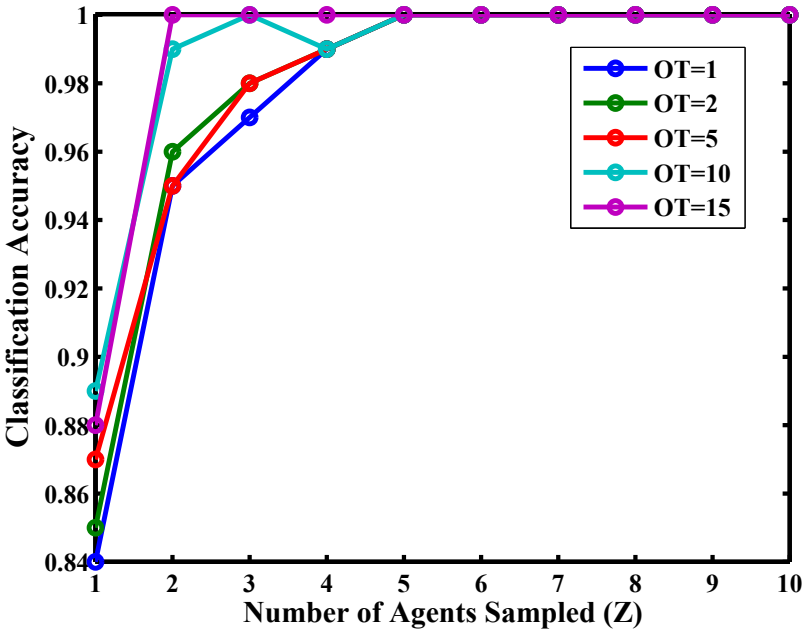


Figure 6.16: Group type classification accuracy as the number of agents sampled, Z, increases for different observation times, OT. Results are for classifying the group type and rotation using both angular velocity and number of neighbors. N = 100 agents.

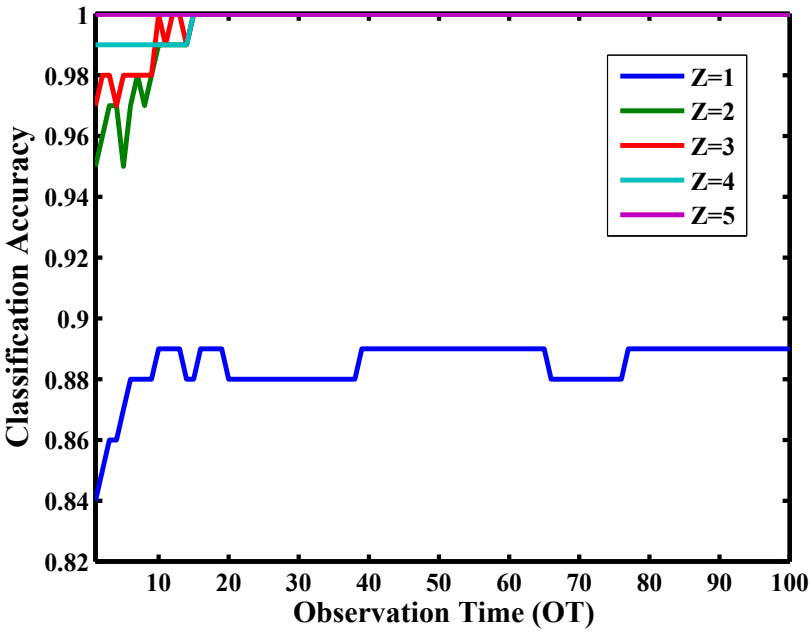


Figure 6.17: Group type classification accuracy as observation time, OT, increases for different numbers of agents sampled, Z. Results are for classifying the group type and rotation using both angular velocity and number of neighbors. N = 100 agents.

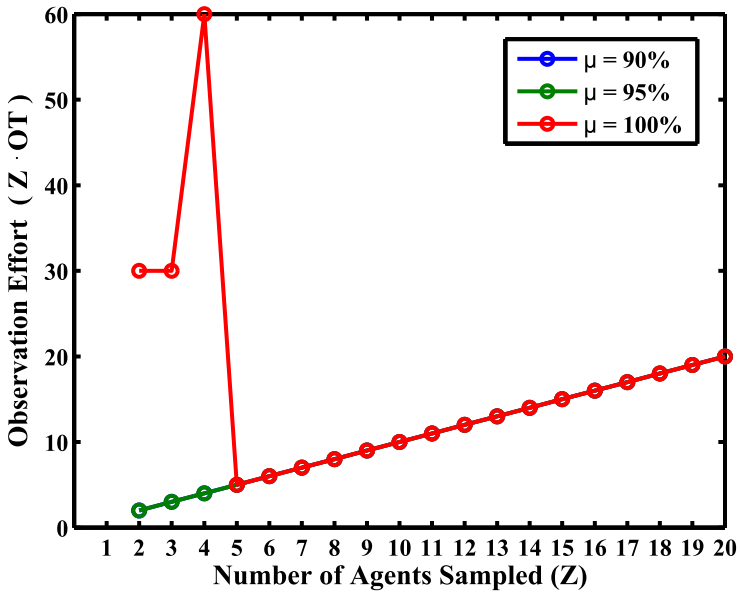


Figure 6.18: Observation effort when classifying group type and rotation using angular velocity and number of neighbors sampled from Z agents and for different accuracy thresholds μ. Observation for $\mu = 90\%$ and $\mu = 95\%$ are identical. Results are for N = 100 agents.

The observation effort is shown in Figure 6.18. The observation effort shown in Figure 6.18 is also interesting. The observation effort is identical for $\mu = 90\%$ or $\mu = 95\%$, and for $Z \geq 5$ there is no difference between the observation effort for any of the three thresholds examined. As with our other classifiers that used number of neighbors, observation effort for Z=1 is undefined. Table 6.7 summarizes the values of Z that minimized observation effort for the different thresholds μ. Interestingly, we see that the observation effort required

	$\mu = 90\%$	$\mu = 95\%$	$\mu = 100\%$
Minimum OE	2	2	5
Corresponding Z	2	2	5

Table 6.7: Values of Z that minimized observation effort (OE) for $\mu = 90$, 95, and 100% accuracy on the test set using angular velocity and the number of neighbors as features.

to classify three group types using angular velocity and number of neighbors with $\mu = 95\%$ is actually lower than the observation effort for classifying two group types using only the number of neighbors. Comparing the results in Table 6.7 with the previous results shown

in tables 6.2, 6.5, and 6.6, we see that using the number of neighbors and angular velocity achieves the lowest observation effort for group type classification.

6.11 Neglect benevolence for group type classification

In this section and the following section we investigate how group type classification accuracy changes depending on when we sample from agents. Walker and Lewis use the term neglect benevolence [37] to denote the benefit of waiting for a swarm to stabilize before interacting with the swarm. Because the torus and flock group types exhibit transient behaviors and take time to fully form, this section investigates whether our model exhibits neglect benevolence when classifying the collective behavior of a swarm.

Group momentum and group polarization are good indicators of the current behavior of a swarm, but they don't tell us anything about group spatial structure. For instance, a flock will have a high polarization and low momentum regardless of whether or not it has a long tail. Additionally, a torus may have a high momentum and low polarization before agents have spaced out evenly around its perimeter. To investigate these properties we calculate the *coverage gap* [36] and *group elongation* [39] of the swarm. The coverage gap of a swarm is a measure of the dispersion of the agents around the group centroid. The coverage gap is defined as the largest angular gap in the swarm relative to the group centroid. The details of how to compute the coverage gap are shown in Algorithm 1. When the coverage gap is minimized we know the torus is fully formed because agents will be maximally dispersed around the torus perimeter. The group elongation, ζ_g, of a swarm is defined as

$$\zeta_g(t) = \max_{i,j} \|c_i(t) - c_j(t)\|. \tag{6.8}$$

This tells us the length of the smallest bounding square that will fully enclose the swarm.

To allow the group types of our model to fully form, we ran 100 simulations for 400 seconds with random initial headings and positions. We visually checked all the simulations

Algorithm 1 Coverage gap calculation

Input: the agent positions, *positions*
$c_g \leftarrow \frac{1}{N}\sum_i positions_i$
$r \leftarrow (positions - c_g)$
coverageGap $\leftarrow 0$
for $i = 1 : N$ **do**
  $angle(i) = \text{atan2}(r(i,2), r(i,1))$
end for
$angle \leftarrow sort(angle)$
for $i = 1 : N - 1$ **do**
  if $|angle(i) - angle(i+1)| >$ coverageGap **then**
    coverageGap $\leftarrow |angle(i) - angle(i+1)|$
  end if
end for
if $|angle(N) - angle(1) - 2\pi| >$ coverageGap **then**
  coverageGap $\leftarrow |angle(1) - angle(N) + 2\pi|$
end if
return coverageGap

	Mean	Standard Deviation	Max
Flock Elongation	14.8854 units	1.6211 units	19.2876 units
Torus Elongation	31.3510 units	1.0317 units	34.9779 units
Flock Gap	0.3177 rad ($\approx$ 18.2 deg)	0.0767 rad ($\approx$ 4.4 deg)	0.5403 rad ($\approx$ 31 deg)
Torus Gap	0.2575 rad ($\approx$ 14.8 deg)	0.0548 rad ($\approx$ 3.1 deg)	0.4538 rad ($\approx$ 26 deg)

Table 6.8: Statistics for group elongation and coverage gap for a fully formed flock and torus.

and found that every simulation had converged to either a flock or a torus by the end of the simulation and that there were no tails in any of the flock simulations. We then used the final state of each simulation to compute the average group elongation and coverage gap for a flock and a torus. Out of the 100 simulation runs there were 59 torus formations and 41 flock formations that formed from random initial conditions. The means and standard deviations for the group elongation and coverage gap are shown in Table 6.8.

We used the maximum flock and torus elongation and the maximum flock and torus coverage gap after 400 seconds as the upper bound on when a group has fully formed. Using this information we computed how long it took each flock and torus simulation to fully form from random starting conditions by measuring the amount of time it took before the group

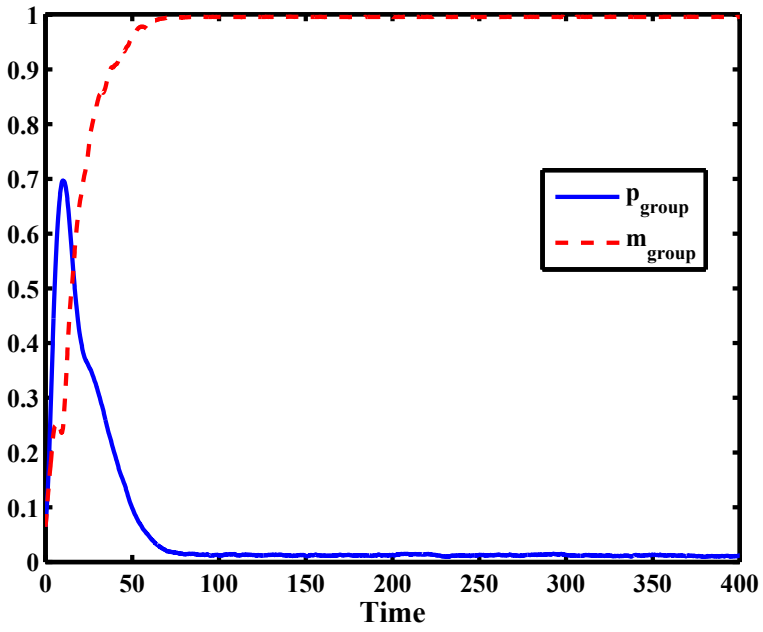
	Mean	Standard Deviation	Max
Flock Elongation Stabilization Time	204.75 seconds	72.39 seconds	339.2 seconds
Torus Elongation Stabilization Time	41.31 seconds	11.3 seconds	81.8 seconds
Flock Gap Stabilization Time	41.32 seconds	42.28 seconds	168.2 seconds
Torus Gap Stabilization Time	28.22 seconds	14.26 seconds	70.1 seconds

Table 6.9: Statistics for the time required for a flock or a torus to fully form when using either the group elongation or the group coverage gap as a measure of how well the group has formed. The times reported were calculated by finding the first time that the group elongation or coverage gap was below the corresponding maximum value reported in Table 6.8.

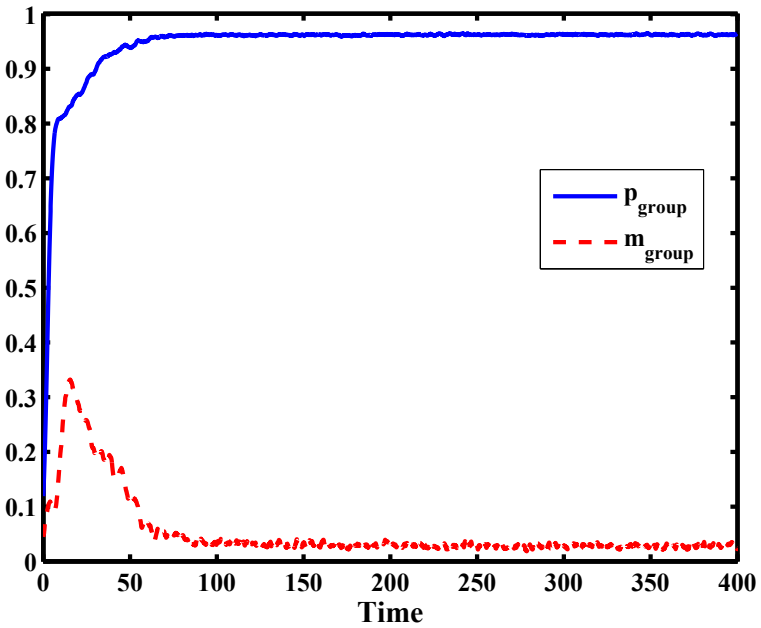
elongation or coverage gap dropped below the upper bound. Table 6.9 shows the results for group elongation stabilization time and coverage gap stabilization time. We waited 10 seconds before checking the coverage gap and elongation so that the effect of the initial placement was mitigated.

The results in Tables 6.8 and 6.9 show that the torus elongation is about twice as large as the flock elongation when both are fully formed. We also see that it takes almost 5 times longer for a flock to fully form than for a torus, when started from random initial conditions. As discussed in Section 6.6.3, this is because a flock typically has long tail of agents trying to catch up to the main body of the flock. Because of the natural movement of agents in the flock, the main body of the flock moves slower than an individual agent so agents in the tail eventually catch up to the flock if given enough time.

We also investigated how p_{group} and m_{group} change over time for the flock and the torus. The average p_{group} and m_{group} over time is plotted in Figure 6.19. Figures 6.20 and 6.21 also show the average group elongation and coverage gap for the same simulations. We see in Figure 6.19 that the torus and flock formations form quickly, usually in about 50 seconds, in terms of m_{group} and p_{group}. However, looking at figures 6.20 and 6.21 we see that the torus elongation stabilizes after about 50 seconds and the coverage gap stabilizes after



(a) Torus



(b) Flock

Figure 6.19: Average group angular momentum (m_{group}) and group polarization (p_{group}) for a flock and a torus started from random initial conditions.

about 75 seconds. As noted before, the time until the flock elongation stabilizes is much longer than for a torus.

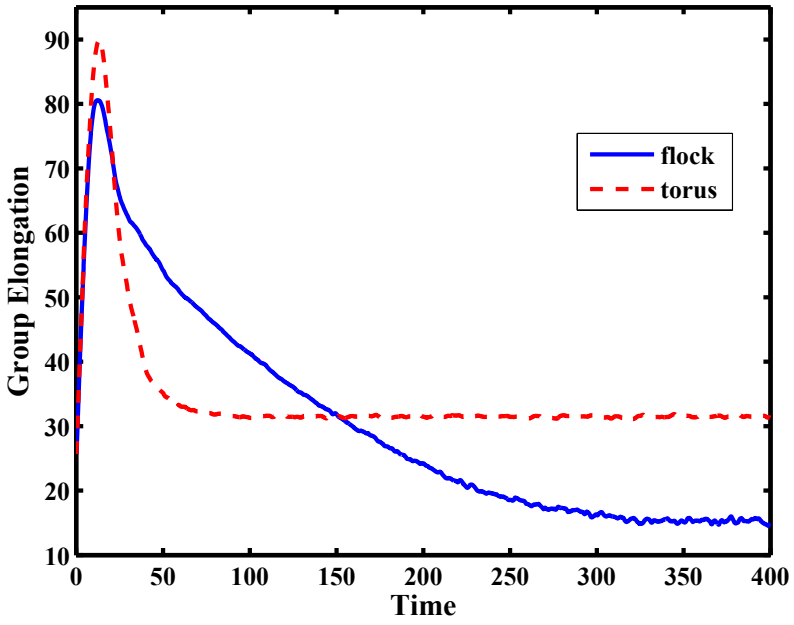


Figure 6.20: Average group elongation over time for a flock and a torus started from random initial conditions.

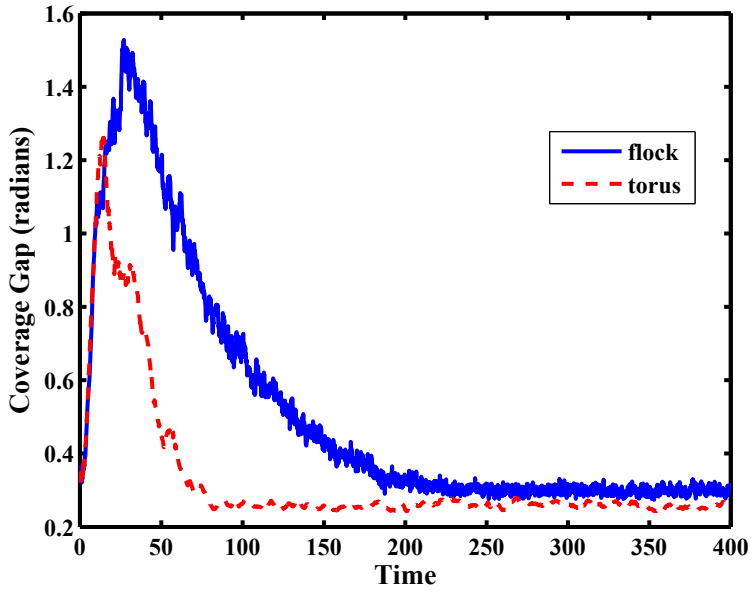


Figure 6.21: Average group coverage gap over time for a flock and a torus started from random initial conditions.

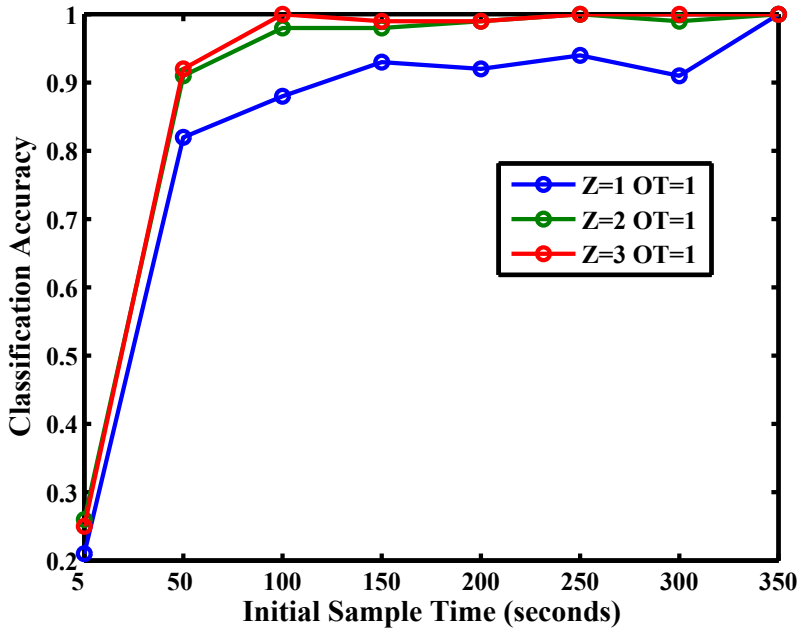


Figure 6.22: Neglect benevolence for classifying between clockwise torus counterclockwise torus and flock using one sample ($OT = 1$) of the number of neighbors and the angular velocity from Z agents. Sample Time represents the simulation time at which the samples were taken, or in other words the time before any samples were taken.

Based on these results, we decided to compare classification accuracies for waiting to sample until after 5, 50, 100, 150, 200, 250, 300, and 350 seconds of simulation time have passed. We used the classifier discussed in Section 6.10 to classify both the group type and rotation of the swarm using both the number of neighbors and the angular velocity as features.

The classification results are shown in Figure 6.22. We see that our classifier does exhibit neglect benevolence as shown by the general increase in accuracy as the initial sampling time increases. Accuracy increases dramatically between initial sample times of 5 and 50 seconds. This corresponds with our earlier analysis that distinct group types are usually formed after 50 seconds, although flocks will often still have long tails. Additionally, when samples were taken after waiting 350 seconds, just sampling once from one agent was sufficient to achieve 100% accuracy on the test set.

6.12 Classification during transient behavior

The results of the previous section show that allowing a group type to fully form has a large impact on classification accuracy. In this section we investigate more thoroughly how our classifier performs in the presence of transient behaviors. To do this we evaluated the classification accuracy at 5 second intervals from 5 to 100 seconds of simulation time. We started sampling after 5 seconds of simulation time to mitigate the effects of initial positions and headings and to allow the group type to begin forming. This resulted in 20 different classification accuracies that we can compare to see how accuracy changes as the different group types emerge. By 100 seconds all the group types will be formed—flocks will be formed, but will have tails—and as shown in previous sections, we can achieve good classification accuracies.

We used the final version of our classifier (as discussed in Section 6.10) that uses both the number of neighbors and the angular velocity of sampled agents to classify between the clockwise torus, counterclockwise torus, and flock group types. Because of the transient behavior of the swarm at the beginning of the simulation we decided to fix $OT = 1$ and increase the sample size only by sampling from different agents. For an emerging group type, newer samples should contain more information than the previous samples, so we classified the group behavior based only on the most recent samples taken. We compared values of Z equal to 1, 2, 5 and 10.

The classification results are shown in Figure 6.23. We see that the classifier performs extremely poorly when sampling once after only 5 seconds of simulation time. The likely reason for this is that many times an elongated flock-like transient forms and then flips into a torus. While the distribution over number of numbers for the elongated group is similar to a torus, detecting the future rotation is difficult the classifier performs poorly. Also, simulations that actually form flocks tend to start off with a large group elongation (see Figure 6.20)

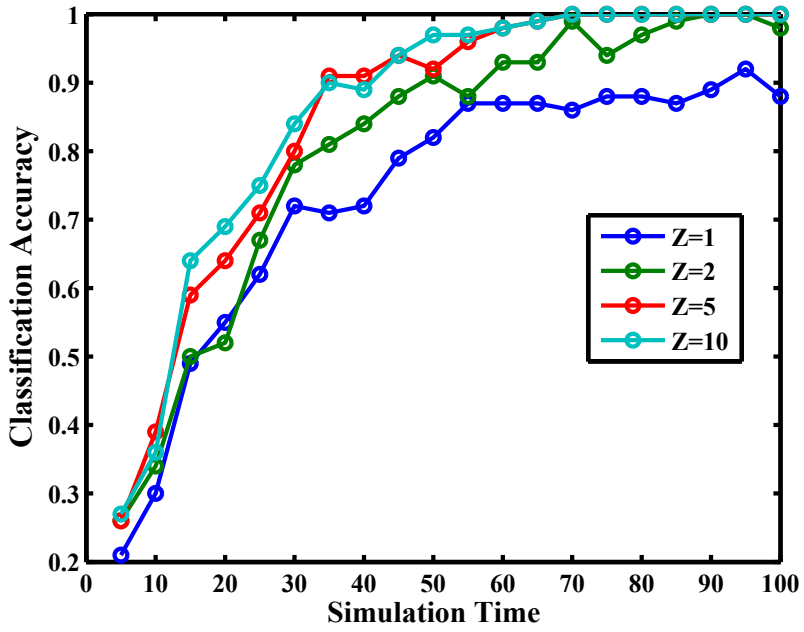


Figure 6.23: Accuracy for classifying between clockwise torus counterclockwise torus and flock using one sample ($OT = 1$) of the number of neighbors and the angular velocity from Z agents.

resulting in agents having fewer neighbors and the group type being initially classified as a torus.

We examined the classification errors for sampling after 10 seconds with $Z = 1$ and found that 89 out of the 100 total simulations were classified as a torus when, in fact, only 41 of the simulations eventually formed into flocks. However, if we wait for 30 seconds before sampling and use $Z = 5$ we see a significant improvement in classification accuracy. Thus, we conclude that while it is highly beneficial to wait for group types to form, group types can be classified with moderate accuracy early in the simulation while only sampling from a small percentage of the agents. The confusion matrices for $Z = 1$ at time 10 and $Z = 5$ at time 30 are included in Appendix B.4.

Number of Neighbors			
N	Group Type	Mean	Standard Deviation
100	Torus	10.1172	2.8638
	Flock	28.9906	7.7746
200	Torus	19.1990	4.1436
	Flock	47.1909	12.1803
300	Torus	27.0929	5.2523
	Flock	62.3854	15.8488
400	Torus	34.3937	6.2799
	Flock	75.9801	19.0570

Table 6.10: The sample mean and standard deviation for the distributions of number of neighbors as the size of the swarm, N, increases.

6.13 Potential scalability to larger group sizes

To investigate the potential scalability of group type classification to larger swarms without actually replicating the previous analyses, we calculated statistics for the number of neighbors and angular velocity of agents in a flock and torus of size $N = 200$, 300, and 400 with $R_o = 8$, $k = .5$ and $s = 5$. 100 replicates were performed for each parameter setting and each simulation was run for 100 seconds. The first 25 seconds were ignored to allow the group to stabilize.

Tables 6.10 and 6.11 show the mean and standard deviation for $N = 100$–400. These results show that the distributions over an agent's number of neighbors and angular velocity continue to be distinct for larger group sizes. Based on these results we hypothesize that accurate group type classification from limited samples is possible with low observation effort for larger groups. Future work should validate this claim.

6.14 Summary

We have demonstrated that the collective behavior of a swarm can be accurately classified using samples from a small percentage of agents in a swarm. Our results show that using a classifier based only on the number of neighbors is sufficient to achieve high classification

Angular Velocity			
N	Group Type	Mean	Standard Deviation
100	Torus	0.3829	0.2001
	Flock	-0.0004	0.6869
200	Torus	0.3805	0.2494
	Flock	-0.0005	0.7225
300	Torus	0.3704	0.3196
	Flock	0.0002	0.7399
400	Torus	0.3601	0.3726
	Flock	0.0001	0.7517

Table 6.11: The sample mean and standard deviation for the distributions of angular velocity as the size of the swarm, N , increases.

accuracy and low observation effort for discriminating between flock and torus group types. Angular velocity works well for discriminating between a flock, a clockwise torus, and a counterclockwise torus; however, using a classifier based only on sampling angular velocity appears to be beneficial only when bandwidth is severely limited. If observing multiple agents is possible, we have shown that combining samples of the number of neighbors with samples of angular velocity achieves higher classification accuracy and lower observation effort on the test set than any other classifier described in this chapter. We showed that our classifier is robust to group types that are not fully formed, the inherent noise from our stochastic topology, and the precision loss that occurs when discretizing angular velocity. Additionally, we demonstrated the existence of neglect benevolence when performing group type classification.

Chapter 7

Estimation of Collective State

In Chapter 6 we demonstrated that, with high accuracy and low observation effort, the group type of a swarm can be classified using samples of a few agents' number of neighbors and angular velocity. The classification was done using only features that did not require self localization or any kind of global reference frame. In this chapter we relax that assumption and assume that, in addition to the number of neighbors and angular velocity, we can sample agent positions and headings. Using this new information we propose several ways of estimating the group centroid and group heading of the swarm using a limited number of local samples. We first present results for estimating the group heading and centroid of a flock and then present results for estimating the centroid of a torus. We assume that the group type has already been detected using the methods discussed in the previous chapter.

7.1 Collective state estimation for a flock

Because the flock group type is characterized by agents forming a tight cluster and all moving in the same direction, we can use relatively simple and straightforward estimates of the group centroid and group heading.

7.1.1 Centroid estimation

To estimate the centroid of a flock, we simply find the centroid of the observable agents and use this to calculate $\hat{c}_g$, an estimate for the actual group centroid c_g, as follows

$$\hat{c}_g = \frac{1}{Z} \sum_{i=1}^{Z} c_i. \tag{7.1}$$

We measure the error as

$$error = \|\hat{c}_g - c_g\| \tag{7.2}$$

where c_g is the actual group centroid. To empirically measure this error, we used 100 simulations for 100 seconds each and tracked the error between the estimated centroid from sampling Z agents and the actual group centroid. Each simulation consisted of $N = 100$ agents. We ignored the first 25 seconds to allow the flock to stabilize.

The resulting average errors and 95% confidence intervals are shown in Figure 7.1. These results show that we can almost always get within about 7 units of the actual group centroid by sampling 1 agent and will on average be within 3.3542 units of the actual centroid. If we increase the number of agents sampled to 5 agents we have an average error of 1.4556 units. With 10 agents the error decreases to 1.0429 units. As expected, the error steadily drops off as more agents are sampled; however, it is interesting to note that as the number of agents sampled increases we see diminishing returns in the amount of error reduction per new agent sampled. Thus, we can obtain good group centroid estimates without having to resort to sampling from a large percentage of the group.

7.1.2 Heading estimation

Successive estimates of a flock's centroid give us one way to estimate the velocity and heading of the swarm. In this section we investigate another way of estimating the heading of a swarm using samples of individual agent headings. Following the notation used in [8], we define the

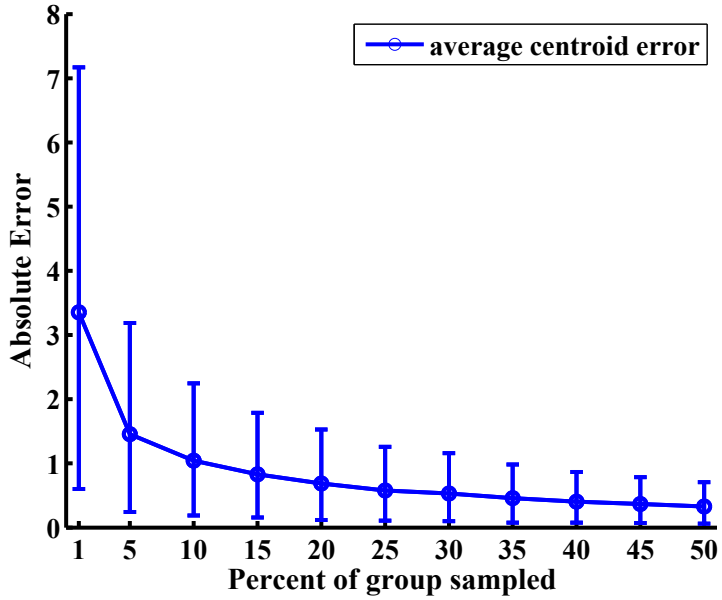


Figure 7.1: Flock centroid error for different values of Z. Results are for 100 replicates of 100 simulation ticks each for a flock formation. Dots represent the average error and error bars represent 95% confidence intervals for the error. $N = 100$ agents.

group direction vector of a swarm as

$$d_g(t) = \frac{1}{N} \sum_{i=1}^{N} v_i(t). \tag{7.3}$$

Using this notation we define the group heading θ_g as

$$\theta_g = atan2(d_g^y(t), d_g^x(t)) \tag{7.4}$$

where $d_g(t) = [d_g^x(t), d_g^y(t)]^T$ is calculated using Equation (7.3).

To estimate (7.4) we simply use the following vector-based average computed from sampled agent velocities/headings as follows

$$\hat{\theta}_g(t) = \text{atan2}\left(\sum_{i=1}^{Z} v_i^y(t), \sum_{i=1}^{Z} v_i^x(t)\right) = \text{atan2}\left(\sum_{i=1}^{Z} \sin(\theta_i(t)), \sum_{i=1}^{Z} \cos(\theta_i(t))\right) \tag{7.5}$$

where $v_i = [v_i^x, v_i^y]^T = [\sin(\theta_i), \cos(\theta_i)]^T$ is the velocity vector of agent i.

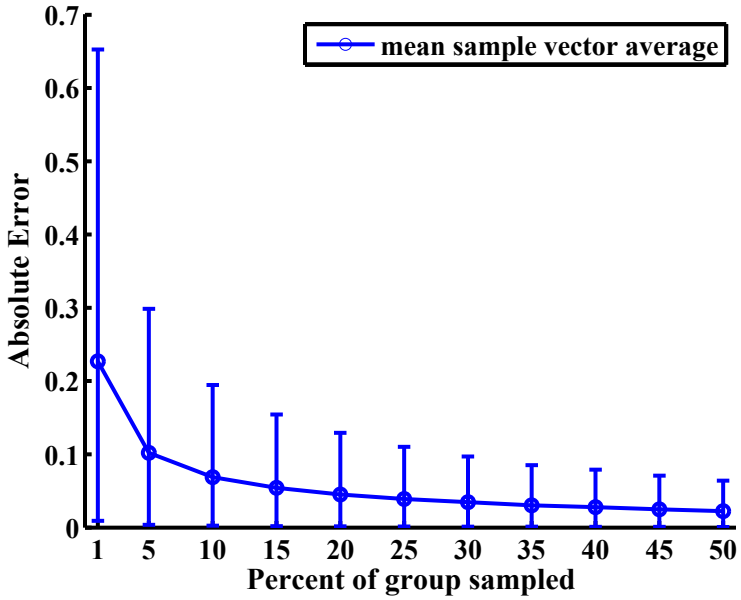


Figure 7.2: Mean absolute error (in radians) between the vector-based estimate of flock heading, $\hat{\theta}_g$, and actual flock heading for different numbers of sampled agents. Error bars show 95% confidence intervals. $N = 100$ agents.

7.1.3 Error analysis for sample group heading

The average error for the vector-based estimator in (7.5) and the 95% confidence intervals for the absolute error between $\hat{\theta}_g$ and θ_g are shown in Figure 7.2. When sampling only one agent there is a mean absolute error of 0.2269 radians (≈ 13.0001 degrees) with a 95% confidence interval of $[0.0091, 0.6526]$ radians ($\approx [0.5237, 37.3915]$ degrees). This large range is due to the noise in individual samples. On the other hand, if we can sample 10% of the swarm (10 agents), the mean absolute error drops to only 0.0687 radians (≈ 3.9381 degrees) and the 95% confidence interval for the error is $[0.0026, 0.1947]$ radians ($\approx [0.1509, 11.1564]$ degrees). If we have enough bandwidth to sample 50% of the flock then the mean absolute deviation between the estimated and actual group headings is only 0.0224 radians (≈ 1.2822 degrees) and the 95% confidence error interval is only $[.0009, 0.0639]$ radians ($\approx [0.0515, 3.6635]$ degrees). These results show that while more bandwidth is always desirable, even with limited bandwidth we can obtain reasonably good estimates of the group heading of a flock as it changes over time.

7.2 Estimation effort for a flock

We want a way to measure the estimation effort required to obtain low error estimates of the group centroid and group heading of a flock. Similar to observation effort, we define estimation effort as

$$EE = Z \cdot ET \tag{7.6}$$

where ET is the *estimation time*, defined as the time to reduce the estimation error below a certain threshold ϵ. However, because we are estimating time-varying quantities, sampling for a longer period of time will not necessarily improve the accuracy of an estimate because older samples will no longer represent good estimates of the current location of the centroid.

We deal with this problem by estimating the parameters of a linear equation for the group centroid c_g as a function of time. If we assume that the centroid is moving along a relatively linear trajectory we can write an equation for $c_g(t)$ as

$$c_g(t) = v_g \cdot t + c_0 \tag{7.7}$$

where $v_g = s_g[\cos\theta_g, \sin\theta_g]^T$ is the group velocity vector with group speed s_g and c_0 is the starting location of the swarm. We now can use least squares to find estimates $\hat{v}_g$ and $\hat{c}_0$ of the parameters v_g and c_0, respectively. Once we have obtained parameter estimates $\hat{v}_g$ and $\hat{c}_0$ we can calculate an estimate of the group heading

$$\hat{\theta}_g = \text{atan2}(\hat{v}_g^y, \hat{v}_g^x). \tag{7.8}$$

We can additionally estimate the the group centroid at some time in the near future using Equation (7.7). Because we are estimating a trajectory we can use successive samples to obtain a more accurate estimate of the centroid and heading of a flock.

In practice, recursive least squares [40] or some other online algorithm could also be used for cases where we need quick recursive updates as we obtain new samples or where the

trajectory is changing and may not be always linear. However, to demonstrate the concept of estimation effort, we simply use linear least squares and assume a linear trajectory hypothesis although other hypotheses could also be used.

Because the trajectory of a flock over a long time period is very random (see Figure 4.5), we simply look at short 100 time step (10 second) intervals. While there is still some random movement, we found that breaking up the trajectory into these smaller intervals allowed us to use a linear approximation that closely approximated the actual trajectory of the flock.

7.2.1 Experiment

As an experiment, we ran 10 simulations with 100 agents starting from random initial conditions but with initial headings all equal to the same random angle between 0 and 2π. We recorded the agent positions and used these positions as noisy estimates of the actual group centroid. These samples were used to find parameters to Equation (7.7) using least squares. We defined the true group heading over the time interval $[t_0, T]$ as

$$\theta_g = \text{atan2}(c_g^y(T) - c_g^y(t_0), c_g^x(T) - c_g^x(t_0)) \tag{7.9}$$

where $T = t_0 + 100$ for our simulations. We then measured the error in predicted group heading and initial group centroid as

$$error_{\theta_g} = |\theta_g - \hat{\theta}_g| \tag{7.10}$$

and

$$error_{c_g} = \|c_g(t_0) - \hat{c}_0\|. \tag{7.11}$$

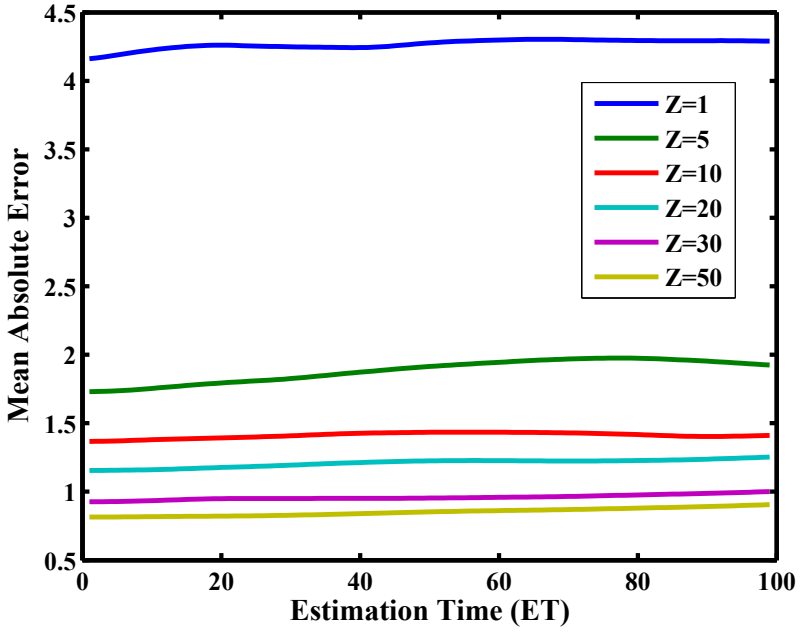
7.2.2 Results

To calculate the estimation error and estimation time, we partitioned each simulation into 10 consecutive 10-second intervals. This resulted in one hundred 10-second intervals over which we computed the accuracy of using linear least squares to estimate the parameters in Equation (7.7). Figure 7.3 shows that increasing Z decreases the average estimation error for both c_0 and θ_g, but with diminishing returns.

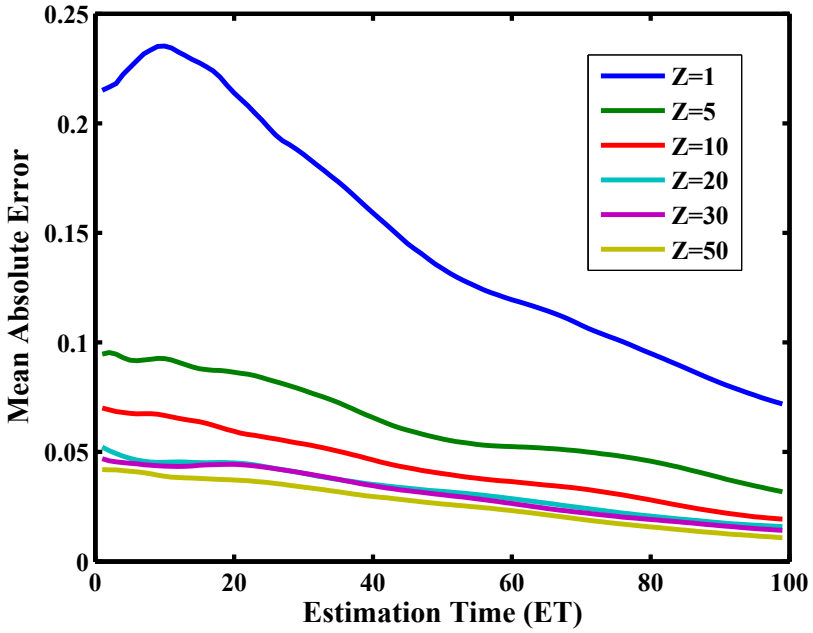
Interestingly, increasing ET causes the estimation error for c_0 to slightly increase on average. We investigated this phenomenon and found that this was because small values of ET correspond to samples closer to c_0, while larger values of ET result in a better fit for the slope of the line (v_g) because the points are dispersed over a wider area but has the effect of slightly increasing the error for c_0.

The initial rise in error in Figure 7.3(b) when $Z = 1$ is due to the fact that a single agent will often move from one side of the swarm to the other in an oscillating pattern causing predictions based on this one agent to sometimes get worse before they get better as more samples are taken. Figure 7.4 shows the same error results, this time as a function of Z. This figure makes clear that sampling from more agents is always better, but that most of the reduction in error happens with the first 20 agents. Thus, sampling from a subset of the agents provides good estimates of the collective state of the swarm.

Figure 7.5(a) shows that, not surprisingly, the estimation effort for estimating the initial group centroid is linear in the number of agents sampled because, as shown in Figure 7.3, increasing the estimation time did not have a significant effect on the initial centroid estimation error. Additionally, we see that the slope of the estimation effort curve for c_0 is 2 because at least two observations are needed to fit a line to the data. Figure 7.5(b) shows that a long estimation time is required to obtain a small group heading error; however, to get an estimate within 5 degrees requires a only a few samples on average.

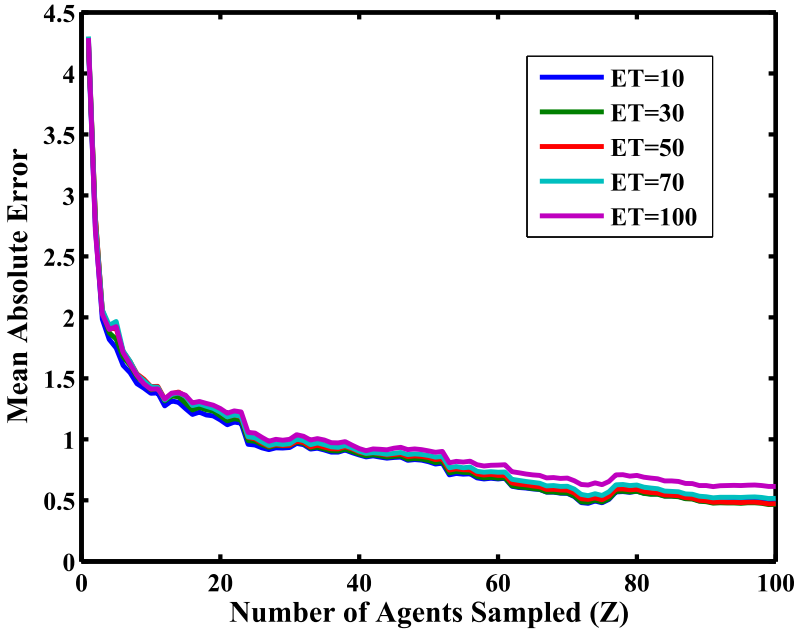


(a) Estimating c_0

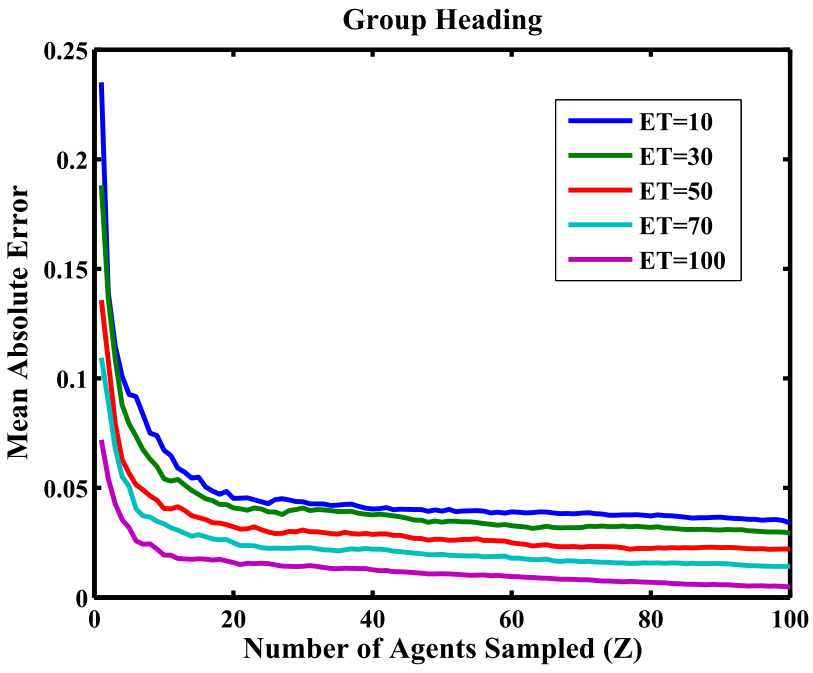


(b) Estimating θ_g

Figure 7.3: Mean absolute error versus ET for estimating c_0 and θ_g for a flock, using linear regression for different values of Z. The error in 7.3(b) in in measured in radians.

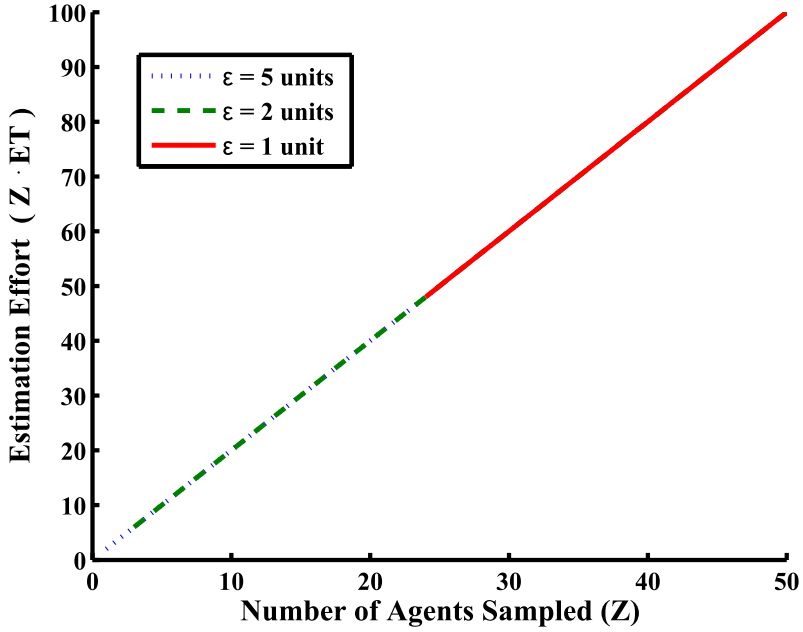


(a) Estimating c_0

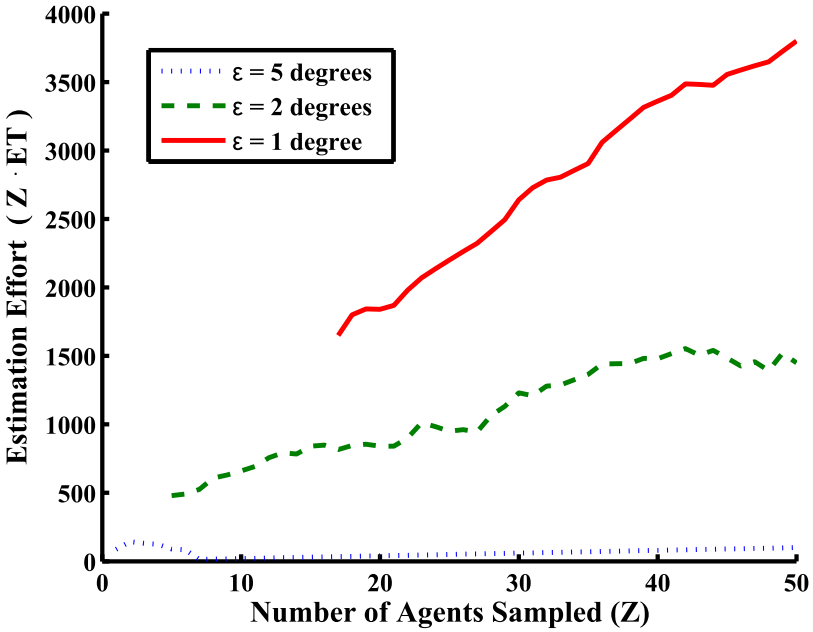


(b) Estimating θ_g

Figure 7.4: Mean absolute error versus Z for estimating c_0 and θ_g for a flock, using linear regression for different values of ET.



(a) Estimating c_0



(b) Estimating θ_g

Figure 7.5: Estimation effort to get the estimation error for c_0 and θ_g below the threshold ϵ.

7.3 Collective state estimation for a torus

The two key global properties that we want to estimate for a torus are the group centroid and the group rotation.

7.3.1 Group rotation

We define the group rotation as

$$rotation_g = sgn\left(\sum_{i=1}^{N} sgn(\omega_i)\right) \tag{7.12}$$

where

$$sgn(x) = \begin{cases} -1, & \text{if } x < 0 \\ 0, & \text{if } x = 0, \\ 1, & \text{if } x > 0. \end{cases} \tag{7.13}$$

Thus, $rotation_g = -1$ means that the majority of the group is turning to the right and has a clockwise rotation. If $rotation_g = 1$, then the majority of the group is turning to the left and has a counterclockwise rotation. Finally, if $rotation_g = 0$, then the group rotation is undefined. We assume that the torus rotation has already been determined using the classifier described in Section 6.10.

7.3.2 Centroid estimation

The group centroid for a torus is more complicated to estimate than for a flock. Given a sample of agent positions we can calculate a naive estimate of the group centroid by averaging the sampled agent positions

$$\hat{c}_g = \frac{1}{Z}\sum_{i=1}^{Z} c_i. \tag{7.14}$$

However, this will only be accurate if enough agents are sampled and if the agents are well spaced around the torus perimeter. If we sample a small number of agents that are close

together this will result in an inaccurate estimate. To better estimate the centroid of a torus from limited samples we offset the positions of sampled agents by a vector orthogonal to their velocity where the projection direction is determined by an agent's angular velocity.

To find the vector orthogonal to the agent velocity, we need to rotate the current velocity vector by $\pm\pi/2$ depending on the angular velocity of the agent. The rotation matrix $R(\theta)$ will rotate a vector by θ and is given by

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{7.15}$$

Thus, given an agent's heading θ with $\omega > 0$ (counterclockwise rotation) the vector orthogonal to the agent's heading in the estimated direction of the centroid is

$$R\left(\frac{\pi}{2}\right) \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} \tag{7.16}$$

and for $\omega < 0$ (clockwise rotation) we have

$$R\left(-\frac{\pi}{2}\right) \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix}. \tag{7.17}$$

Thus, to estimate the centroid of the torus we simply multiply the unit orthogonal vector by the estimated torus radius $\hat{r}$ and add this vector to the agent's position giving an estimate of the centroid for agent i as

$$\hat{c}_g^i = c_i + \mathrm{sgn}(\omega_i) \cdot \hat{r} \cdot \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix} \tag{7.18}$$

where the sign of ω takes care of the issue of rotation. If we sample from more than one agent, we can combine multiple estimates to get the following estimate of the group centroid

$$\hat{c}_g = \frac{1}{Z}\sum_{i=1}^{Z} \hat{c}_g^i. \tag{7.19}$$

7.3.3 Torus radius

Based on our theoretical analysis performed in Section 4.3.2, we know that the steady state radius of a torus with attraction and orientation dynamics is $4s/k\pi$. Using this as our estimated torus radius we have the following estimate of the group centroid

$$\hat{c}_g^i = c_i + \text{sgn}(\omega_i) \cdot \frac{4s}{k\pi} \cdot \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix} \tag{7.20}$$

where $\text{sgn}(\omega)$ insures that the vector will be projected in the correct direction.

7.3.4 Error analysis

To see the difference between using the fixed-radius estimate (7.20) and the naive centroid estimate (7.14), we used 100 simulations that were initialized to form a torus. We left out the first 25 seconds to allow the torus to fully form. We then calculated the mean absolute error for each method using different numbers of sampled agents. The mean absolute errors for the fixed radius estimates are shown in Figure 7.6 along with the error of the naive centroid estimates. We see that the naive centroid estimate performs better than the fixed radius estimate only if more than 85% of the swarm is sampled. When sampling fewer than 20% of the swarm, there is a large difference in error between the two estimates, and we see that the radius-based estimate provides centroid estimates that have much lower error even when sampling from a single agent (1% of the swarm). Additionally, sampling 20 or 30% of the swarm using the radius-based estimator yields mean absolute errors that are only slightly

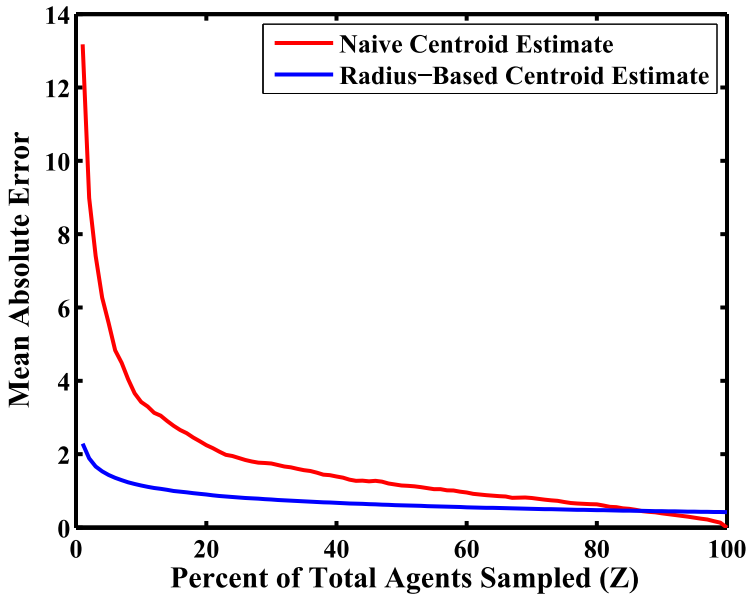


Figure 7.6: Mean absolute error of fixed-radius estimates and naive estimates over samples from 100 simulations leaving out the first 25 seconds for different values of Z. Results for $N = 100$ agents.

larger than the errors obtained by the same estimator when sampling more than 50% of the swarm. Thus, high accuracy estimates of the group centroid are possible using only local samples from a few agents.

7.4 Estimation effort for a torus

In this section we examine the estimation effort for estimating the centroid of a torus. Because the torus centroid is relatively stationary we can simply take successive centroid estimates over time and average them to obtain a better estimate of the group centroid. To investigate the estimation effort for estimating the centroid of a torus, we used the same 100 torus simulations used in the previous section and looked at values of Z between 1 and 50 and values of ET between 1 and 100. We have seen in Figure 7.6 that increasing Z decreases our estimation error. Figure 7.8 shows that increasing estimation time also tends to reduce the estimation error, but that this reduction shrinks as Z is increased. Figure 7.8 shows the estimation effort for estimating the centroid of a torus. We see that estimates within 1 or 2

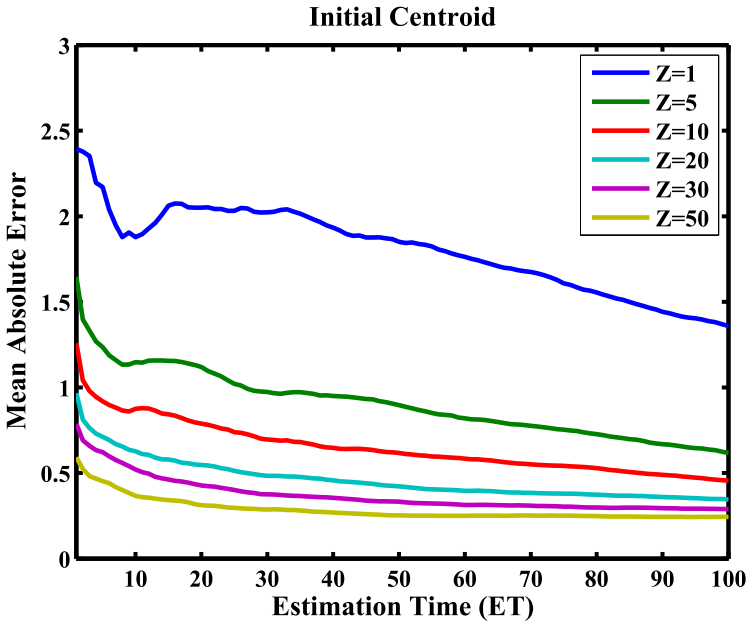


Figure 7.7: Mean absolute error versus estimation time for estimating the centroid of a torus using samples from Z agents. Results are for $N = 100$ agents.

units can be achieved with low estimation effort, but that getting estimates within 0.5 units requires either sampling from many agents or sampling from a few agents for a long time.

7.5 Summary

We have shown that if agent positions and headings can be sampled, then the actual position and heading of the swarm can be estimated. While we did not explicitly estimate the group velocity, we note that the velocity of a swarm can be estimated using successive estimates of the group centroid. We demonstrated that accurate estimates of the group centroid and heading can be obtained using samples from a small percentage of the swarm. We also showed how the concept of estimation effort can be applied to estimating the group heading and centroid of a flock and the group centroid of a torus.

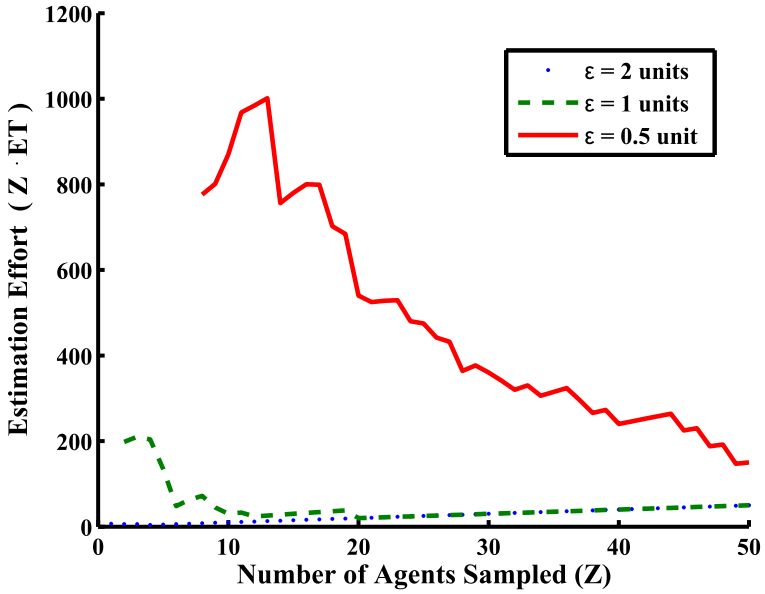


Figure 7.8: Estimation effort required to estimate the centroid of a torus using samples from Z agents with an average error less than the threshold ϵ. Results are for $N = 100$ agents.

Chapter 8

Classifying the Collective Behavior of Other Models

To demonstrate that the methods used for group type classification in Chapter 6 generalize to other multi-agent systems, we consider two additional models of swarming behavior that exhibit distinct collective behaviors caused by local agent interactions. We first consider group type classification for a bio-inspired model proposed by Couzin [8]. We then discuss how group type classification could be applied to a physics-based swarm model [10].

8.1 Couzin's Model

Couzin's model [8] is similar to the swarm model introduced in Section 3.1, but has several key differences: (1) it produces three distinct group types: a swarm, a torus, and a flock, (2) agents have a blind spot, (3) it uses a fixed maximum turning rate instead of using integrator dynamics, (4) it uses non-stochastic and non-overlapping behavior zones to create a metric topology with a maximum sensing range R_a, and (5) it adds explicit noise to the individual agent headings.

The agent dynamics in Couzin's model are similar to ours; agents repel from neighbors within a repulsion radius R_r, orient with agents within an orientation zone R_o, and attract to agents within an attraction zone R_a. The model consists of N agents that move at a constant speed s, have a vision range of α radians, and can turn at most β radians per second. Agents update their heading and position every Δt seconds. We refer the reader to [8] for the model specifics. The simulations reported below use the following simulation parameters $N = 100$, $R_r = 1$, $R_a = 15$, $s = 3$ units/second, $\alpha = 5\pi/3$ radians, and $\beta = 1.4$ radians. We used

$\Delta t = 0.1$ seconds and added $\eta \in \mathcal{U}(-0.2, 0.2)$ radians/second of noise to the individual agent headings, where $\mathcal{U}(\cdot, \cdot)$ is the continuous uniform distribution.

8.1.1 Tipping points

To find parameter settings that can exhibit multiple group types, we performed an experiment similar to the one discussed in Section 4.2, but for Couzin's model. We ran a series of 100 replicates for values of R_o between 1 and 10 in 0.5 unit increments. We then used the results of the final state of the simulation after 100 seconds to determine what group type had formed. Each replicate was started from random initial conditions. We determined the group types as follows

$$type_g = \begin{cases} \text{swarm}, & \text{if } m_{group} < 0.3 \text{ and } p_{group} < 0.3 \\ \text{torus}, & \text{if } m_{group} > 0.6 \text{ and } p_{group} < 0.3 \\ \text{flock}, & \text{if } m_{group} < 0.3 \text{ and } p_{group} > 0.6 \end{cases} \tag{8.1}$$

Because the group types in Couzin's model do not form as clearly as in our model, these conditions are more relaxed than the conditions for distinguishing between a flock and a torus found in Section 3.19.

Figure 8.1 shows the probability of each group type forming for each value of R_o that was simulated. We see that there is not a tipping point between the swarm and torus group types. This is because once the radius of orientation is greater than the radius of repulsion, the agents start orienting with each other and form a torus. This can be seen by the immediate jump in the probability of forming a torus and the immediate drop in the probability of forming a swarm when R_o is increased from 1 to 1.5. As the radius of orientation is increased past 1.5, the torus stabilizes and then, for high enough values of R_o, flocks become the dominant group type. We see that $R_o = 6$ is the approximate tipping point between the torus and flock formations.

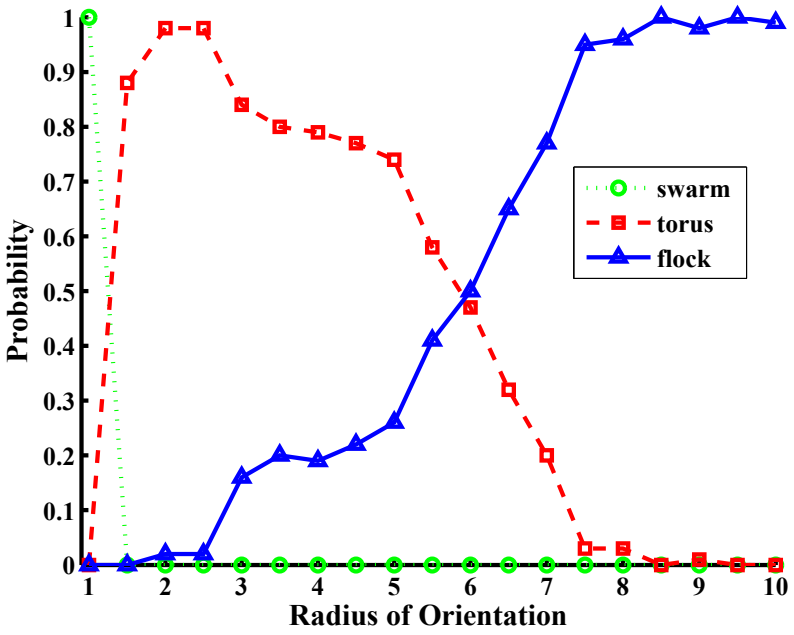


Figure 8.1: Probability of Couzin's model producing a swarm, a torus, and a flock for different values of R_o.

Figure 8.2 shows the percentage of simulations that converged to one of the group types for different values of R_o. This figure shows that some of the simulations never formed into a swarm, torus, or flock. We examined the simulations that didn't meet any of the criteria in Equation 8.1 and found that they either formed a noisy torus with agents rotating in the both directions (for $R_o = 1.5$) or formed something in between a flock and a torus, with most of the agents clumped up but still moving around in circles. We also checked for fragmentation by looking at the positions of the agents at the end of each the simulation and checking to see if the underlying interaction graph formed by the radius of attraction was connected. We found that only two of the simulations fragmented. These two simulations were not used in the following analysis.

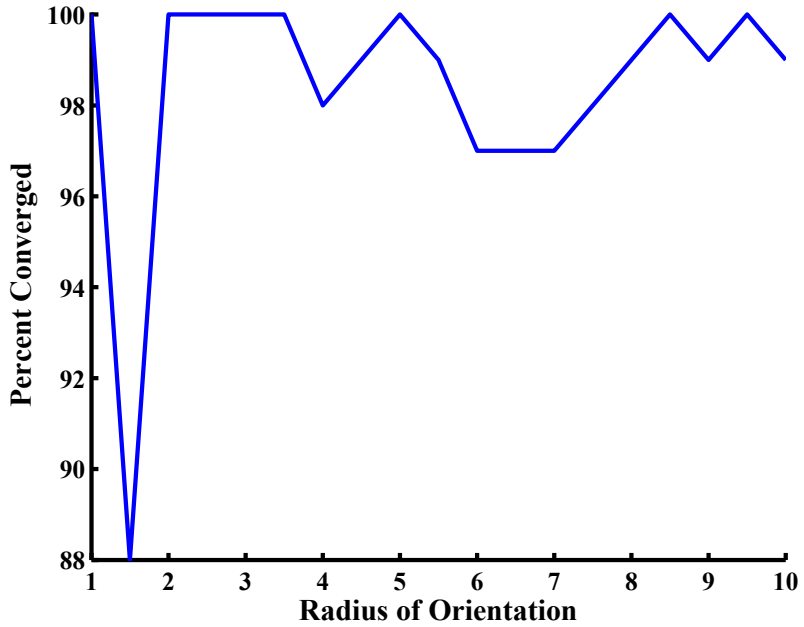


Figure 8.2: The percentage of simulations that converged to either a swarm, torus, or flock.

8.1.2 Classification features

While there are many possible features that could be used in group type classification we chose to use the same features used in Chapter 6, namely, number of neighbors and angular velocity.

We used data from the simulations used to investigate the tipping points to calculate the distributions over angular velocity for a swarm, a torus, and a flock. We used the 100 replicates with $R_o = 1$, the 98 replicates that formed a clockwise or counterclockwise torus when $R_o = 2$, and the 99 replicates that formed a flock and did not fragment when $R_o = 8.5$ to model the swarm, clockwise and counterclockwise torus, and flock, respectively. To determine whether the torus was rotating clockwise or counterclockwise we looked at the final angular velocity of each agent for each replicate. If the majority of the agents were turning counterclockwise we designated the simulation as a counterclockwise torus. If the majority of the agents were turning clockwise we designated the simulation as a clockwise

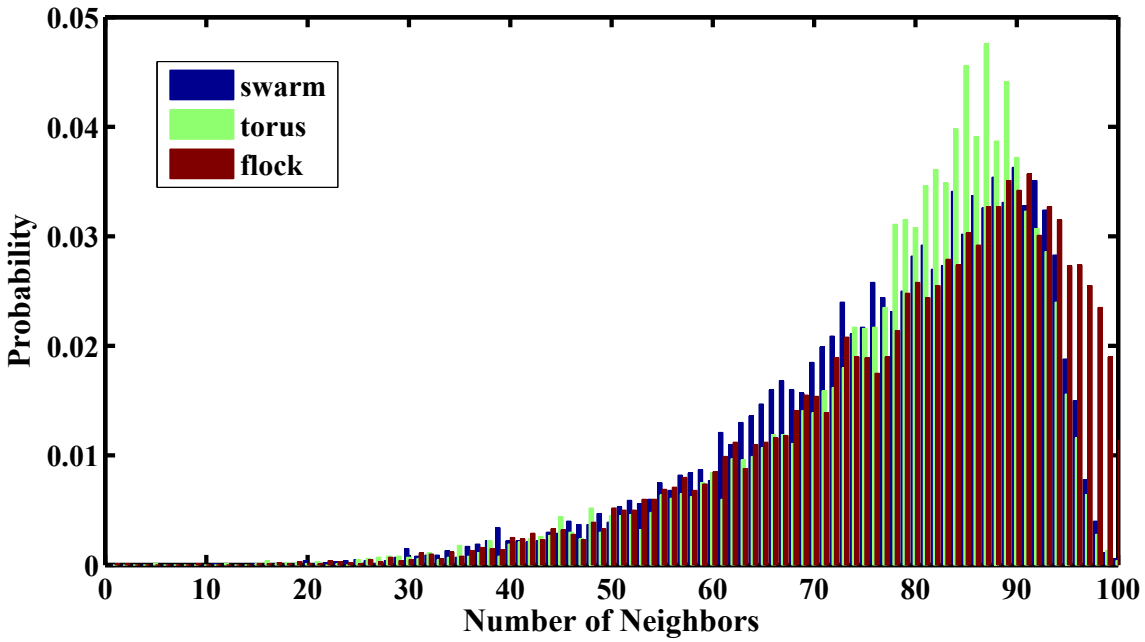


Figure 8.3: Distributions over an agent's number of neighbors in a swarm, torus, and flock formed by Couzin's model.

torus. We found that there were 47 clockwise torus simulations, and 51 counterclockwise torus simulations.

Figure 8.3 shows the distribution over number of neighbors in a swarm, torus, and flock. We see that the distributions in Couzin's model are similar. This is because all three group types tend to be tightly packed with the torus lacking the large void that is found in our model. There are, however, some slight differences between the distributions. Figure 8.4 shows discretized distributions over angular velocity in a swarm, clockwise torus, counterclockwise torus, and flock. Once again we see that there is not as clear a distinction in Couzin's model as in our model, but that there are still salient differences.

8.1.3 Torus and flock classification

We created a test set of 100 simulations using $R_o = 6$ to test the accuracy of a naive Bayes classifier using the number of neighbors and angular velocity as features. Using $R_o = 6$ will

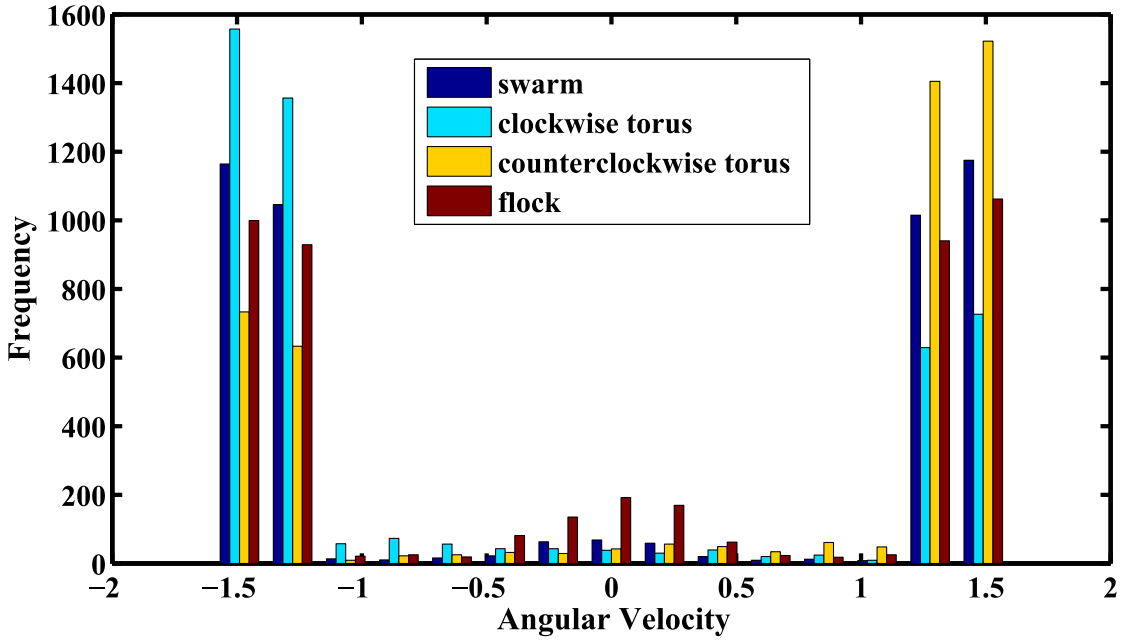


Figure 8.4: Distributions over an agent's angular velocity (radians/second) in a swarm, torus, and flock formed by Couzin's model.

form either a flock or a torus with approximately equal probability so we have one of three possible group types: a clockwise torus, a counterclockwise torus, or a flock. Figures 8.5 and 8.6 show how the classification accuracy changes as the number of agents sampled and the observation time changes.

We see that the classification accuracies for Couzin's model (shown in Figure 8.5) are much lower than for our model (see Figure 6.16), but that sampling more than 10 agents with enough observation time provides very high classification accuracies. The lower accuracies are to be expected given the distributions of the features; however, the results do show that an accuracy above 80% is achievable when sampling from a small number of agents.

We calculated the observation effort for thresholds $\mu = 90\%$, $\mu = 95\%$, and $\mu = 100\%$. These results are shown in Figure 8.7. Table 8.1 summarizes the values of Z that minimized observation effort for the different thresholds μ. We see that more observation effort is needed

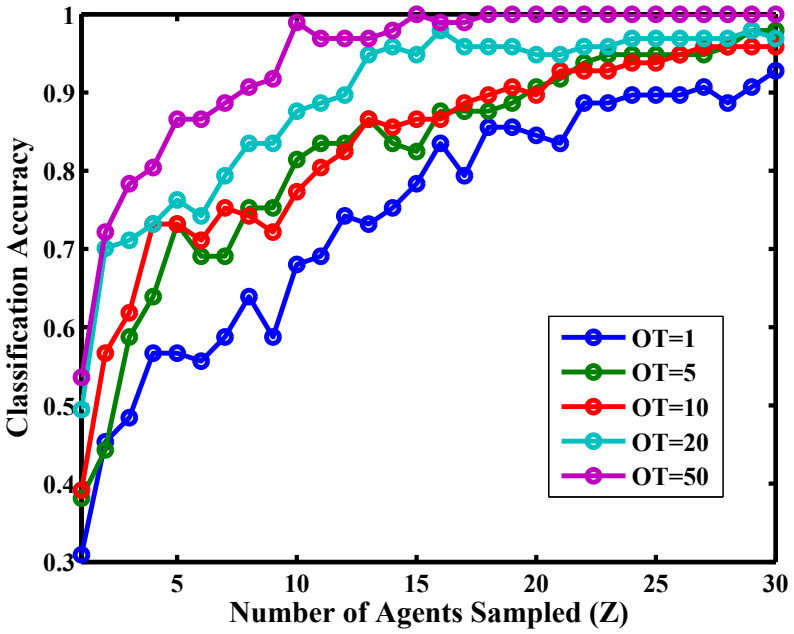


Figure 8.5: Classification accuracy for Couzin's model using an agent's number of neighbors and angular velocity as features. Results are for distinguishing between a flock, a clockwise torus, and a counterclockwise torus.

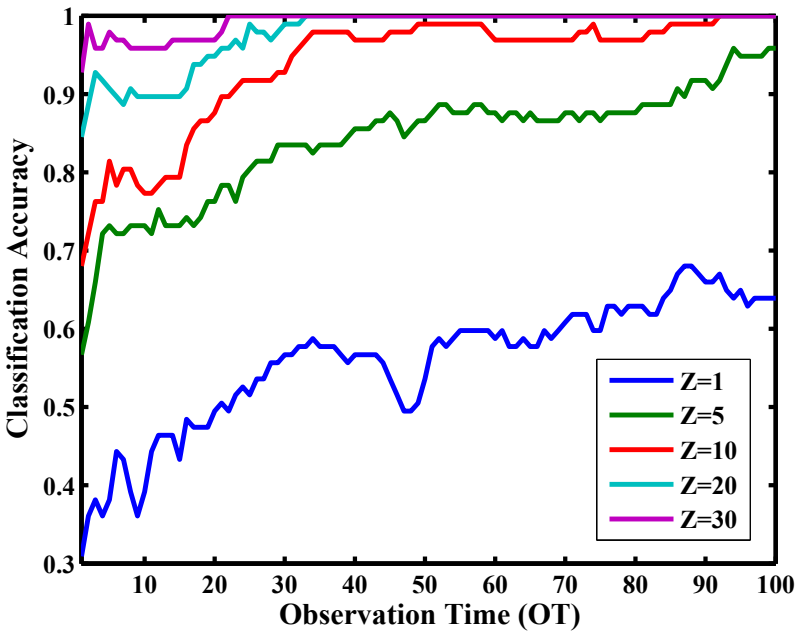


Figure 8.6: Classification accuracy for Couzin's model using the number of neighbors and the angular velocity sampled as a function of OT. Results are for distinguishing between a flock, a clockwise torus, and a counterclockwise torus.

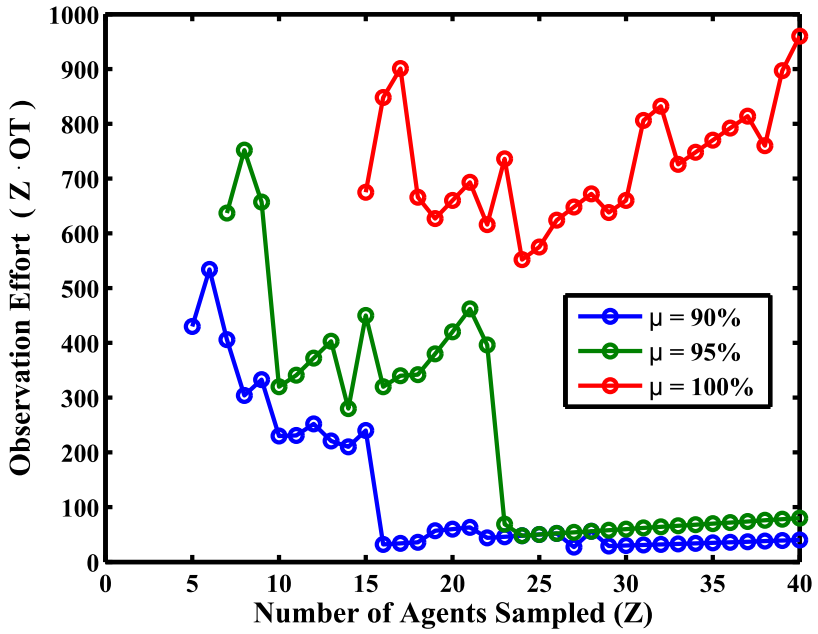


Figure 8.7: Observation effort calculated for Couzin's model using the number of neighbors and the angular velocity sampled from Z agents.

	$\mu = 90\%$	$\mu = 95\%$	$\mu = 100\%$
Minimum OE	27	48	552
Corresponding Z	27	24	24

Table 8.1: Values of Z that minimized observation effort in Figure 8.7 for $\mu = 90$, 95, and 100% accuracy on the test set.

to classify group types in Couzin's model than in our model (compare with Table 6.7), but we also see that 100% accuracy on the test set can be achieved by sampling less than 30% of the swarm if the observation time is long enough.

Error analysis

We found that the low classification accuracy of 30.9% for $Z = 1$ and $OT = 1$ was because most of the flock simulations were misclassified as either a clockwise or counterclockwise torus. Increasing the number of samples by letting $Z = 5$ and $OT = 5$ results in a dramatic increase in accuracy to 73.2%, but the majority of the errors still come from misclassifying the flock group type as a torus (see Appendix B.5 for more details). Interestingly, when $Z = 5$ and

$OT = 5$, there are no misclassifications between the clockwise and counterclockwise torus. Thus, our chosen features discriminate well between torus types, but do not discriminate as well between a flock and a torus. Future work should investigate whether different features provide better group type classification and lower observation effort for Couzin's model.

8.1.4 Swarm, torus, and flock classification

Couzin's model has a tipping point between the torus and flock group types, but not between the swarm and torus group types. Because of this there is no single set of parameters that will form either a swarm or torus with equal probability. However, it is still interesting to see if group type classification can be done with high accuracy when the swarm can be in more than three group types. To test the classification accuracy in this case, we created a test set with 100 simulations of a swarm ($R_o = 1$), 100 simulations of a torus ($R_o = 2$), and 100 simulations of a flock ($R_o = 8.5$). The values of R_o were chosen to ensure a high likelihood of every simulation forming the desired group type. All other parameters were kept the same as before, and each simulation was started with random initial positions and headings.

Before computing the accuracy of the classifier we checked each simulation's final m_{group} and p_{group} to see if the desired group type had formed. We also checked the final positions of the agents to check for group fragmentation. We found that none of the simulations fragmented for any of the parameter settings. For $R_o = 1$, all 100 of the simulations formed a swarm. For $R_o = 2$, 93 of the simulations formed a torus, 4 simulations formed a flock, and 3 simulations failed to converge to a distinct group type. For $R_o = 8.5$, 98 of the simulations formed a flock and 2 never formed a distinct group type. In the subsequent analysis we use only the simulations that fully formed a group type. This results in a test set of 295 simulations.

Figure 8.8 shows classification accuracy as a function of Z for several different values of OT. We see that the classification accuracies with three group types are lower than with two possible group types. As before, we see that more samples increases accuracy, and that

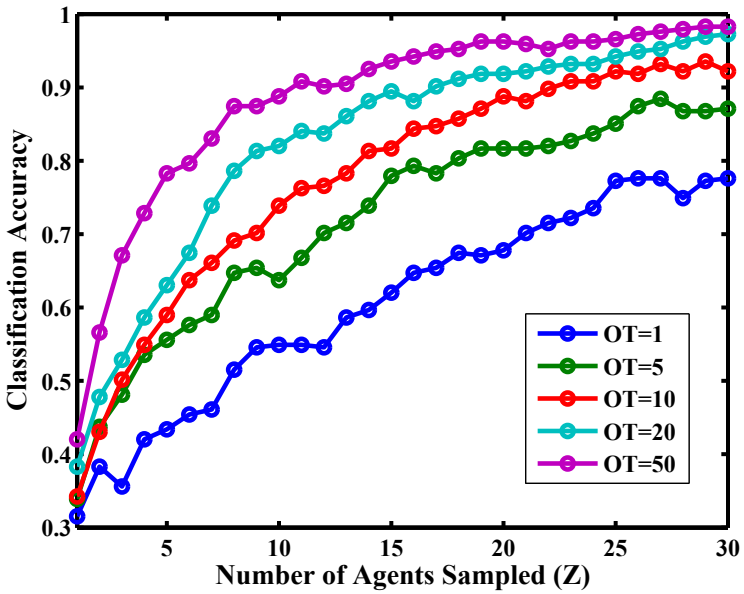


Figure 8.8: Classification accuracy as a function of Z using angular velocity and number of neighbors as features. Results are for distinguishing between a swarm, a clockwise torus, a counterclockwise torus, and a flock.

	$\mu = 85\%$	$\mu = 90\%$	$\mu = 95\%$
Minimum OE	36	80	205
Corresponding Z	40	24	41

Table 8.2: Values of Z that minimized observation effort in Figure 8.10 for $\mu = 90$, 95, and 100% accuracy on the test set.

more samples are needed to reach high classification accuracies with Couzin's model than for classification using our model.

Figure 8.9 shows classification accuracy as a function of OT. We see that accuracy tends to increase as the observation time is increased, but never reaches 100% on the test set for values of $Z \leq 30$. Figure 8.10 shows the observation effort for classifying between four group types. Because of the lower classification accuracies, we computed the observation effort using the thresholds $\mu = 85\%$, $\mu = 90\%$, and $\mu = 95\%$. Table 8.2 summarizes the values of Z that minimized observation effort for the different thresholds μ. From these

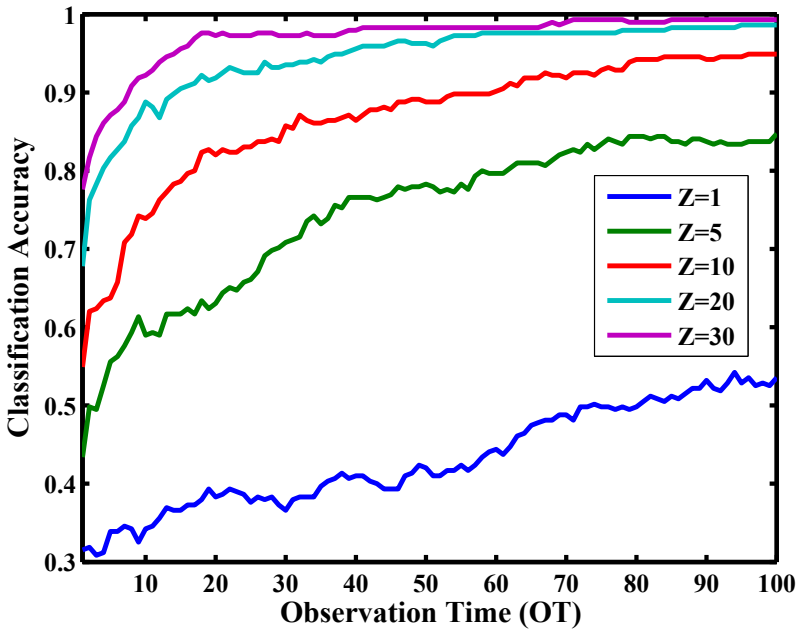


Figure 8.9: Classification accuracy as a function of OT using angular velocity and number of neighbors as features. Results are for distinguishing between a swarm, a clockwise torus, a counterclockwise torus, and a flock.

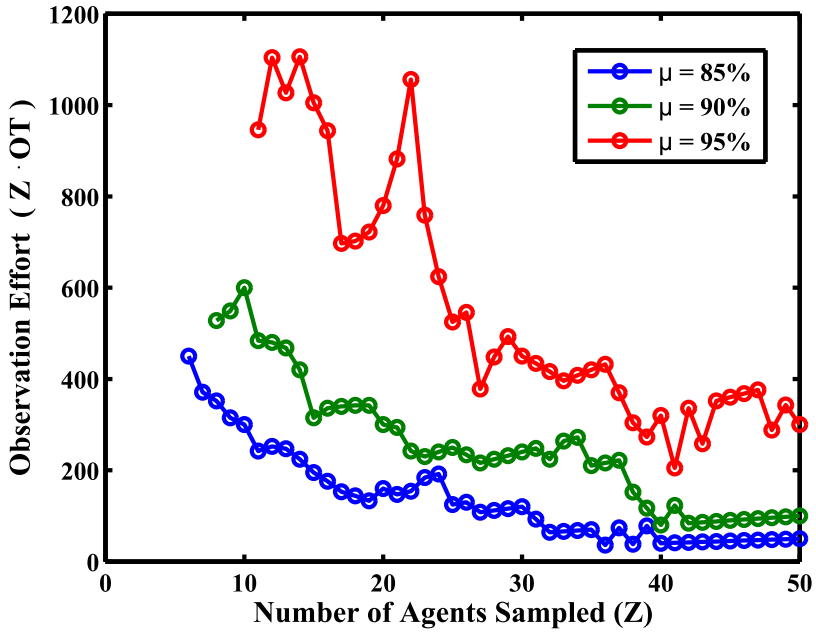


Figure 8.10: Observation effort required to classify the swarm, clockwise torus, counterclockwise torus, and flock group types in Couzin's model for different classification accuracy thresholds.

results we see that four group types can be classified with reasonable accuracy if enough agents are sampled. However, we see that increasing the number of group types leads to higher observation effort and lower accuracies overall.

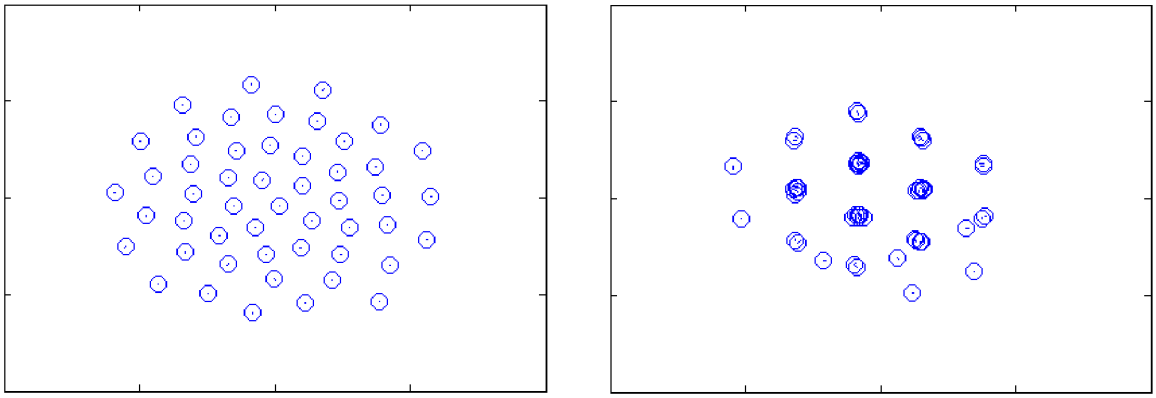
Error analysis

We examined the classification errors for a couple of values of Z and OT (see Appendix B.6 for details) and found that most of the errors came from from misclassifying a group type as a swarm or misclassifying a swarm as another group type. We also found that the largest number of these errors occur between the flock and swarm group types. This reflects the similarity between the distributions for the number of neighbors and angular velocity of a flock and of a swarm as seen in figures 8.3 and 8.4.

8.2 Physicomimetics model

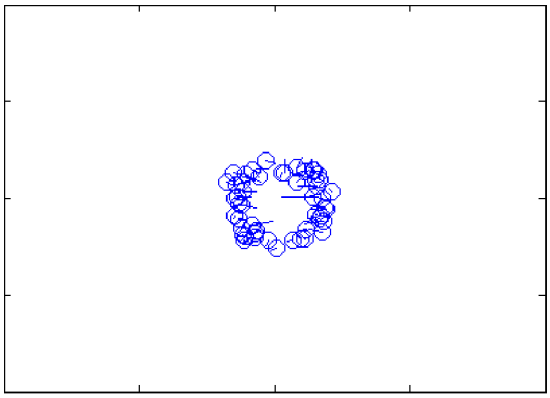
In this section we briefly discuss how group type classification could be applied to a model based on physics rather than biology. Physicomimetics is a distributed control law that uses physics-based forces to control a swarm [10]. The physicomimetics model is fundamentally different from our model and Couzin's model in several ways: (1) agents only attract and repel from one another, (2) agents can accelerate, (3) agents have no blind spot, (4) the model includes friction, (5) there is no limitation on the acceleration or velocity of the agents.

Previous work in our lab has identified three different group types that form based on different parameters: isotropic, anisotropic, and structured [41]. Snapshots of these collective structures are shown in Figure 8.11. We see that, while these group types are different than the the group types exhibited by our model and by Couzin's model, there are still fundamental global differences between these group types. We hypothesize that the collective group types shown in Figure 8.11 can be estimated from local samples of individual agents. However, different features than the ones previously examined in this thesis would likely be



(a) Isotropic

(b) Anisotropic



(c) Structured

Figure 8.11: Three different group types in a physicomimetics particle swarm model.

needed to obtain accurate classification. Based on the spatial structure of the group types, features that appear promising include the minimum distance between any two agents, group elongation, a measure of how well the sampled agent positions fit a circle, agent acceleration, and local agent density.

8.3 Discussion

In Chapter 6 we demonstrated that highly accurate group type classification was possible using only local samples from a few agents in the swarm. In Chapter 7 we showed that low error estimates of the group centroid and heading of a swarm can be achieved using local samples of position and heading from a small subset of the agents in the swarm. To show that these ideas are applicable to other swarm models, this chapter has examined two additional models of swarming: Couzin's model [8] and the physicomimetics model [10]. Both models can produce several distinct group types depending on the model parameters and initial conditions. For Couzin's model, we have shown that accurate group type classification is possible using samples of local information from a small number of agents. While we did not actually perform classification on the physicomimetics model, we described several potential features for classifying group types in this model. We hypothesize that most self-propelled particle models that produce distinct group types will be amenable to collective behavior classification and estimation techniques similar to those described in this thesis.

Chapter 9

Quorum Sensing

This final chapter addresses the goal of increasing the scalability of human-swarm interactions while limiting the vulnerability of a swarm to agent failures. We show that adding a biologically inspired quorum sensing mechanism to a swarm achieves this goal. To facilitate a discussion of quorum sensing, we define a simple taxonomy of agent types. An agent's type determines how it responds to human influence, external influence, and other agents. We divide agents into two classes: *human-aware* agents who can respond to human input and *human-blind* agents who do not respond to human input. In this chapter we explore the properties of a swarm in which we have both human-aware agents and human-blind agents. Specifically, we investigate a particular kind of human-aware agent called a *stakeholder* and a particular kind of human-blind agent called a *type-aware* agent [42].

9.1 Stakeholders

Stakeholders are influenced by both the human and by other agents. Each stakeholder has a priority parameter $\rho \in [0, 1]$ that determines the priority of human influence over influence from other agents. If ρ is high, then the stakeholder is responds more to human influence than to its neighbors. If ρ is low, then the stakeholder is influenced more by its neighbors than by human influence.

Stakeholders can be led in two different ways: attraction and orientation [36]. Stakeholders that are led by attraction have the desired direction

$$u_i = u_i^{sa} + u_i^o + u_i^r \tag{9.1}$$

where

$$u_i^{sa} = \frac{\rho\hat{q}_i + (1-\rho)u_i^a}{\|\rho\hat{q}_i + (1-\rho)u_i^a\|}, \tag{9.2}$$

$$\hat{q}_i = \frac{q - c_i}{\|q - c_i\|}, \tag{9.3}$$

$q \in \mathbb{R}^2$ is a reference input generated by the human, $\rho \in [0, 1]$ is priority parameter, and u_i^a, u_i^o, u_i^r are the usual attraction, orientation, and repulsion influences described previously in Equations (3.7),(3.8), and (3.9). Stakeholders that are led by orientation have the desired direction

$$u_i = u_i^a + u_i^{so} + u_i^r \tag{9.4}$$

where

$$u_i^{so} = \frac{\rho\hat{q}_i + (1-\rho)u_i^o}{\|\rho\hat{q}_i + (1-\rho)u_i^o\|}. \tag{9.5}$$

Thus, the only difference between leading a stakeholder by attraction and leading a stakeholder by orientation is whether the human influences a stakeholder's attraction vector, u_i^a or orientation vector, u_i^o. It is important to note that an agent is only considered a stakeholder if it is receiving human input. Thus, throughout the remainder of this thesis, when we use the word stakeholder we refer to an agent that is currently being influenced by the human. Following the notation used in [36] we use M to denote the number of stakeholders, i.e., the number of agents receiving human influence.

9.2 Type-aware agents

Type-aware agents are influenced by both human-aware agents and human-blind agents, but are not influenced by the human. Type-aware agents have an awareness parameter $\alpha \in [0, 1]$ that determines the degree of type awareness of the agent. If α is high, then the type-aware agent is influenced more by human-aware agents than by other human-blind agents. If α is low, then the type-aware agent is influenced more by other human-blind agents and less by human-aware agents. For the remainder of this chapter we assume that in the quorum sensing swarm model, agents are either stakeholders or type-aware agents. Thus for a swarm of size N with M stakeholders, we have $N - M$ type-aware agents.

Type-aware agents follow the normal agent dynamics described in Section 3.1 unless one or more of their neighbors are stakeholders. We use quorum sensing to adjust the awareness parameter α_i for each type-aware agent i in the swarm. Quorum sensing is used in biological systems to regulate the emergence of different behaviors depending on external thresholds [43][44]. To incorporate quorum sensing we define a quorum threshold Q_i on the number of neighbors of type-aware agent i that are stakeholders. If type-aware agent i has more than Q_i stakeholder neighbors, then it will temporarily increase its type awareness by setting $\alpha_i = \alpha_i^{max}$, and maintain this increased type awareness until the number of stakeholders within its sensing range falls below Q_i, at which point the agent sets $\alpha_i = \alpha_i^{min}$. We use α_i^{max} and α_i^{min} to denote the maximum and minimum awareness for agent i. The parameter α_i^{max} determines how much agent i is influenced by stakeholder neighbors if it senses a quorum. The parameter α_i^{min} determines how much agent i is influenced by stakeholder neighbors if it does not sense a quorum. Thus, if we let $\mathcal{N}_i = \{j \mid a_{ij} = 1\}$ represent the set of neighbors of agent i, and $\mathcal{S}$ be the set of stakeholders in the swarm, then for any type-aware agent i we have

$$\alpha_i = \begin{cases} \alpha_i^{max}, & \text{if } Q_i < |\mathcal{N}_i \cap \mathcal{S}| \\ \alpha_i^{min}, & \text{if } |\mathcal{N}_i \cap \mathcal{S}| \leq Q_i. \end{cases} \tag{9.6}$$

The desired direction vector of type-aware agent i is

$$u_i = \frac{u_i^{ta}}{\|u_i^{ta}\|} + \frac{u_i^{to}}{\|u_i^{to}\|} + u_i^r \tag{9.7}$$

where

$$u_i^{ta} = \alpha_i \frac{\sum_{j \in \mathcal{N}_i \cap \mathcal{S}} (c_j - c_i)}{\| \sum_{j \in \mathcal{N}_i \cap \mathcal{S}} (c_j - c_i) \|} + (1 - \alpha_i) \frac{\sum_{j \in \mathcal{N}_i \setminus \mathcal{S}} (c_j - c_i)}{\| \sum_{j \in \mathcal{N}_i \setminus \mathcal{S}} (c_j - c_i) \|} \tag{9.8}$$

and

$$u_i^{to} = \alpha_i \frac{v_i + \sum_{j \in \mathcal{N}_i \cap \mathcal{S}} v_j}{\| v_i + \sum_{j \in \mathcal{N} \cap \mathcal{S}} v_j \|} + (1 - \alpha_i) \frac{v_i + \sum_{j \in \mathcal{N}_i \setminus \mathcal{S}} v_j}{\| v_i + \sum_{j \in \mathcal{N}_i \setminus \mathcal{S}} v_j \|} \tag{9.9}$$

and $\mathcal{N}_i \setminus \mathcal{S}$ represents the set of all non-stakeholder neighbors of agent i and u_i^r is given by Equation (3.7). Thus, the type-aware agent's attraction and orientation vectors are weighted between neighbors that are stakeholders and neighbors that are not stakeholders by the parameter α_i.

The effect of increasing α is to increase the influence stakeholders have over type-aware agents, thereby amplifying human influence over the swarm. Because this amplification only happens when there are Q_i or more stakeholders in a type-aware agent's sensing neighborhood, the threshold Q_i acts as a nonlinear switch that increases the responsiveness of the swarm once enough agents are influenced by the human. We demonstrate empirically that this nonlinear switch based on the threshold Q_i increases the scalability of human influence over a swarm by decreasing interaction effort. We also demonstrate empirically that the embedded quorum response limits the vulnerability of the swarm to agent failure.

9.3 Switching between attractors under human influence

To demonstrate the increased scalability and limited vulnerability afforded by quorum sensing we consider the task of using stakeholders to change the collective behavior of the swarm. In particular we consider using human influence to switch group types. In this section we use the quorum sensing swarm model (i.e., a swarm made up of stakeholders and type-aware

agents) to switch from a flock to a torus and from a torus to a flock. We compare the results of leading the stakeholders by attraction and leading the stakeholders by orientation using different numbers of stakeholders, M, and using different stakeholder priorities, ρ.

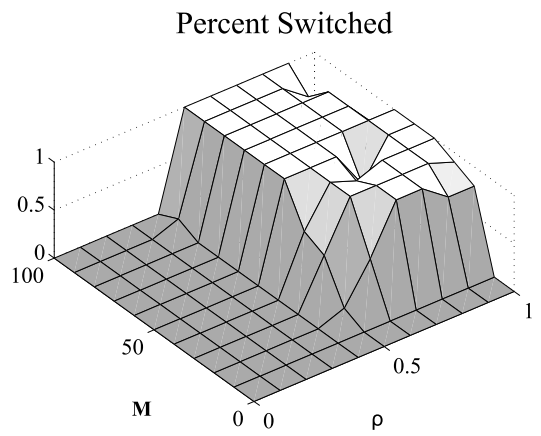
We ran 10 simulations for $M = 10$ to 100 in increments of 10 and $\rho = 0.1$ to 1 in increments of 0.1, and gave each simulation 25 seconds to stabilize before human influence was applied. We gave the swarm 200 seconds to switch group types, then removed the human influence and gave the swarm 50 seconds to stabilize. For each simulation, we calculated the percentage of swarms that switched given the specific number of stakeholders receiving human influence, M, and stakeholder priority value, ρ, and also the percentage of trials that stayed switched after the human influence was removed.

To limit the number of parameters in these simulations we set $\alpha_i^{min} = 0$, $\alpha_i^{max} = 1$, and $Q_i = 0$ $\forall i$—we investigate the effect of higher quorum thresholds later in this chapter and refer the reader to Appendix C.2 for a sensitivity analysis over different values of α^{max} and α^{min}. We calculated the number of trials that switched by simply keeping track of the group angular momentum and polarization throughout the simulation. The group type of the swarm was determined using the following definition

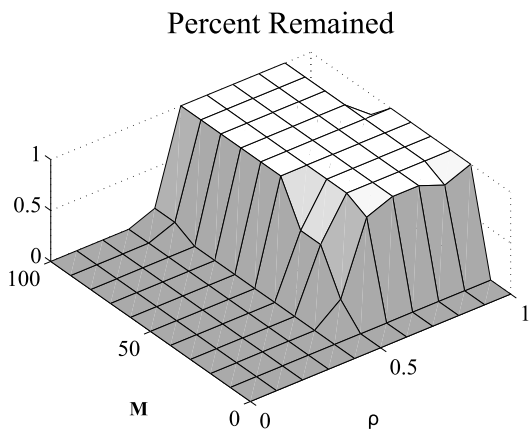
$$type = \begin{cases} torus, & \text{if } p_{group} < 0.25 \text{ and } 0.75 < m_{group} \\ flock, & \text{if } m_{group} < 0.25 \text{ and } 0.75 < p_{group}. \end{cases} \tag{9.10}$$

After 200 seconds of human influence, if the desired group type has formed, as defined by (9.10), then we counted that simulation as a successful switch. If the swarm remained in the new group type for 50 seconds after the human influence was released, then we counted that simulation as having successfully remained.

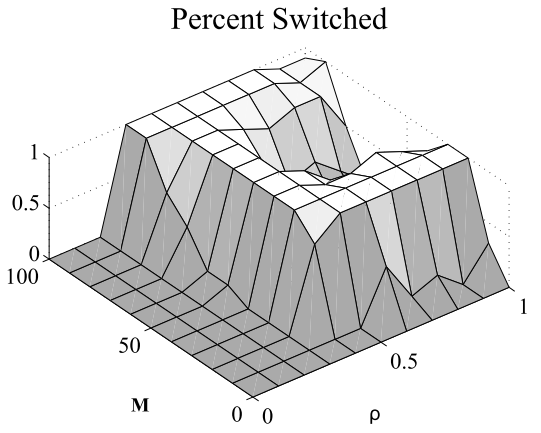
When switching from a flock to a torus, all agents were given random initial positions and initial headings $\theta_i = 0$, $\forall i$. After allowing the group to stabilize for 25 seconds, the constant input $\hat{q} = c_g(25) + [0, 10]^T$ was applied to each stakeholder to cause the group to turn and form a torus. When switching from a torus to a flock, all agents were given random



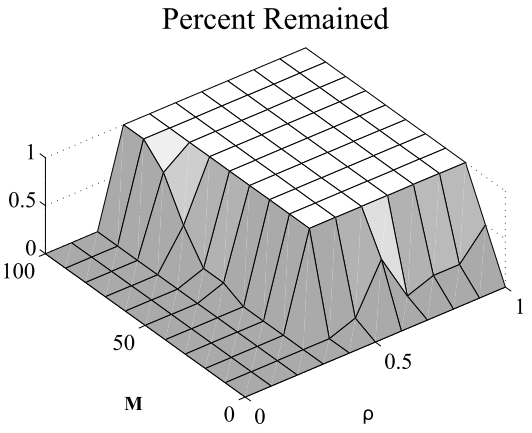
(a) Lead by attraction



(b) Lead by attraction



(c) Lead by orientation



(d) Lead by orientation

Figure 9.1: Switching from flock to torus using quorum sensing.

initial positions and initial headings $\theta_i = \text{atan2}(c_i^x, c_i^y) + \pi/2$ to form a counterclockwise torus. After letting the group stabilize, an arbitrarily large constant control input, $\hat{q} = [10,000, 0]^T$, was applied to the stakeholders to cause them to form a flock. The results of switching from flock to torus and from torus to flock where stakeholders are led by attraction and led by orientation are shown in figures 9.1 and 9.2.

9.3.1 Switching from flock to torus

We see that when switching from a flock to a torus (see Figure 9.1), leading stakeholders by attraction causes the group to switch consistently for $0.5 < \rho < 1$ and $20 < M < 100$. There is a noticeable drop in the number of simulations that switched and remained for $\rho = 1$ and high values of M. We investigated this and found that the agents tended to form a single tightly packed clump that rotated the desired torus centroid. Because the agents never spread out into a full torus, when the control input was removed the group returned to a flock formation.

When leading stakeholders by orientation, we found a large discrepancy between the number of simulations that switched to a torus and the number of simulations that remained as a torus (see figure 9.1(c) and 9.1(d)). Examining these simulations, we found that leading stakeholders by orientation often caused stakeholders to rotate in different directions around the desired torus centroid preventing m_{group} from dropping below 0.25. When the control input was removed the agents oriented themselves to rotate in the same direction, resulting in a higher percentage of simulations that remained a torus than switched to a torus. Because of the problems with agents rotating in different directions, we only lead stakeholders by attraction when switching from a flock to a torus for the remainder of this thesis. Future work should examine whether lead by orientation can be adjusted to provide better switching probability while under human influence.

9.3.2 Switching from torus to flock

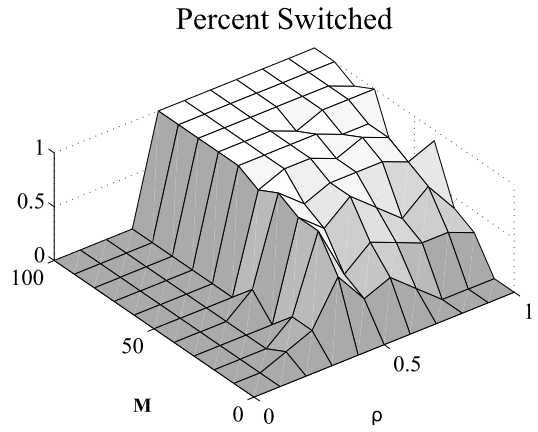
When switching from a torus to a flock (see Figure 9.2), leading stakeholders by orientation worked much better than leading stakeholders by attraction. We examined these results and found that leading stakeholders by attraction was successful in causing the agents to switch from a torus to a flock. However, the attraction input $\hat{q}$ caused the stakeholders to slowly pull away from the rest of the group causing the flock to elongate. Thus, when the human influence was removed, the flock was unstable and usually flipped back on itself and

reformed a torus. Figures 9.2(a) and 9.2(b) show that except for simulations with $\rho = 0.5$, simulations that switched to a flock usually switched back to a torus. Using leadership by orientation eliminated this phenomenon and caused the agents to form a less elongated flock that remained in the flock formation after human influence was removed (see figures 9.2(c) and 9.2(d)). Based on these results we only lead stakeholders by orientation when switching from a torus to a flock for the remainder of this thesis.

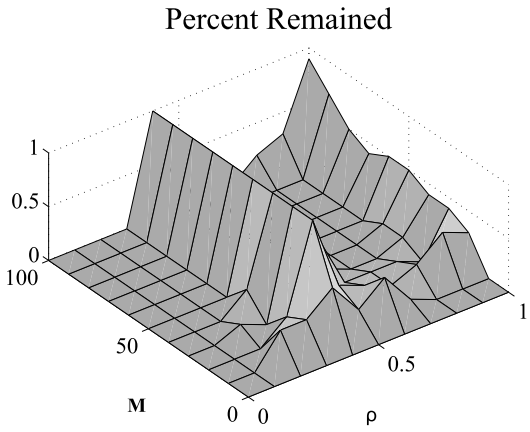
The results for switching from a flock to a torus and from a torus to a flock using the quorum sensing model closely match the results found by Kerman [36] who used only stakeholders—no type-aware agents or quorum sensing—to switch between group types. We will refer to the model used by Kerman as simply the *stakeholder model*.

9.4 Scalability of the quorum sensing model compared to the stakeholder model

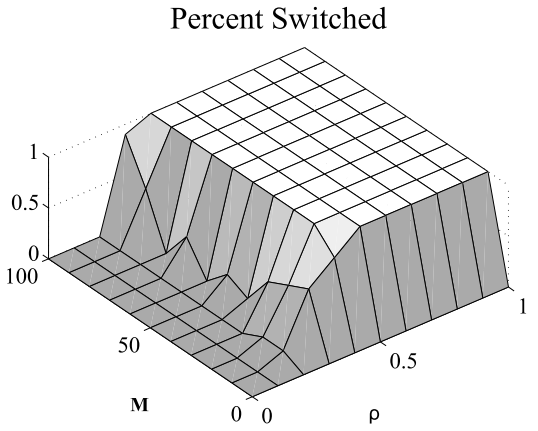
To demonstrate that quorum sensing increases the scalability of human-swarm interactions, we compare the quorum sensing model with the stakeholder model [9], which uses only stakeholders and no type-aware agents. To investigate the scalability of these two models we repeated the above experiment using both the stakeholder model and the quorum sensing model for swarm sizes of $N = 100$, 200, 300, and 400 agents. We used the same parameters as before and varied M from 10 to $N/2$ in 10 agent increments. When switching from torus to flock, we led the M stakeholders by orientation. When switching from flock to torus, we led the M stakeholders by attraction. As the size of the swarm increases, the difference in scalability between the two models is very distinct. Figures 9.3 and 9.4 show plots of the the probability of switching and remaining switched for $N = 400$ (plots of the results for $N = 200$ and $N = 300$ are included in Appendix C.1). In both figures we see that the quorum sensing model can switch group types for much smaller numbers of human-influenced agents, M.



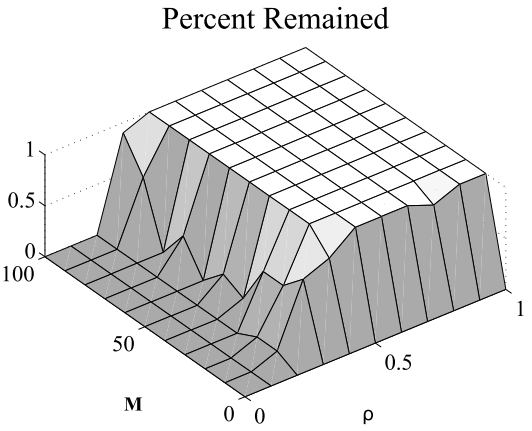
(a) Lead by attraction



(b) Lead by attraction

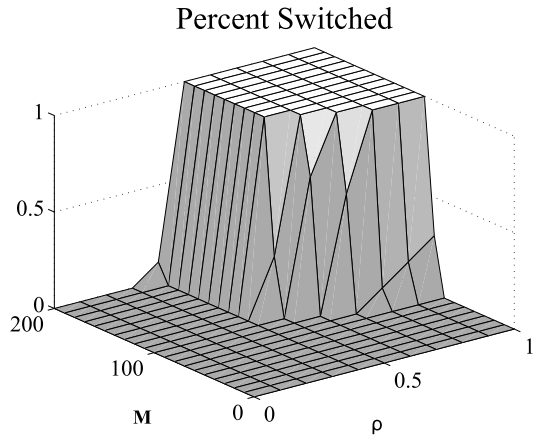


(c) Lead by orientation

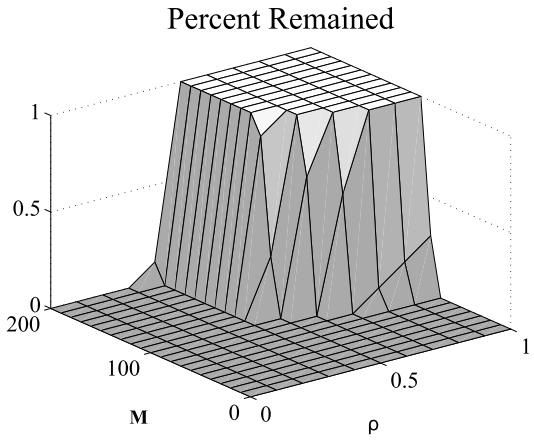


(d) Lead by orientation

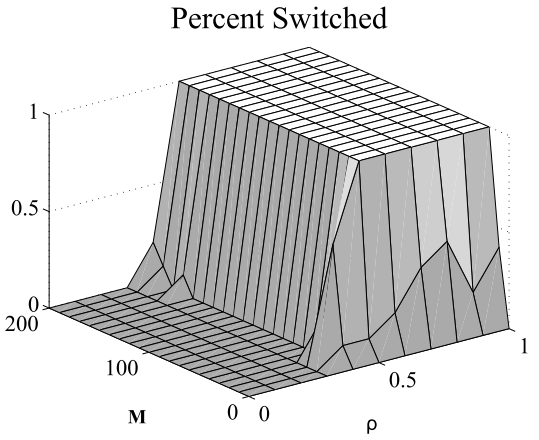
Figure 9.2: Switching from torus to flock using quorum sensing.



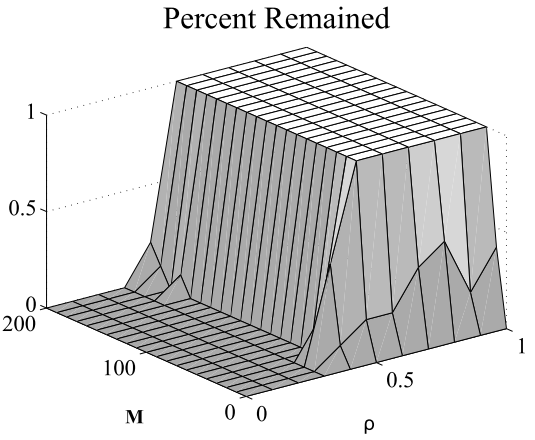
(a) Stakeholder switch from torus to flock



(b) Stakeholder switch from torus to flock

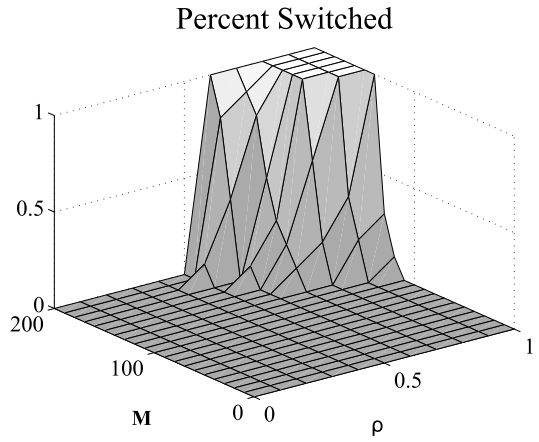


(c) Quorum sensing switch from torus to flock

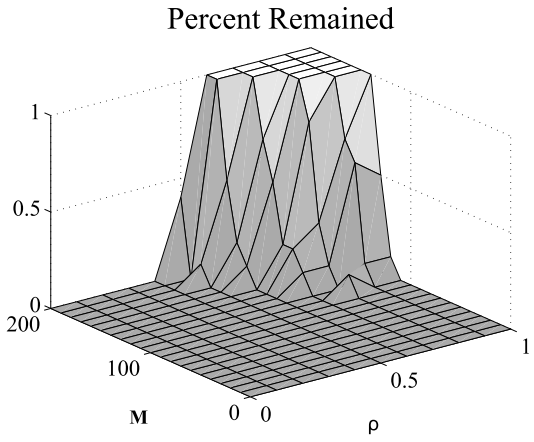


(d) Quorum sensing switch from torus to flock

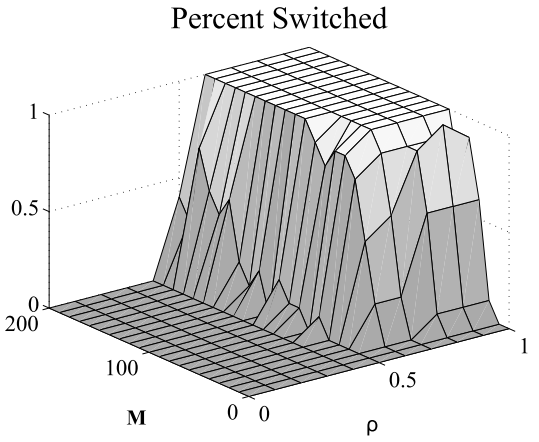
Figure 9.3: Switch from a torus to a flock using the stakeholder model and the quorum sensing model. Stakeholders are led by orientation. Results are shown for N=400.



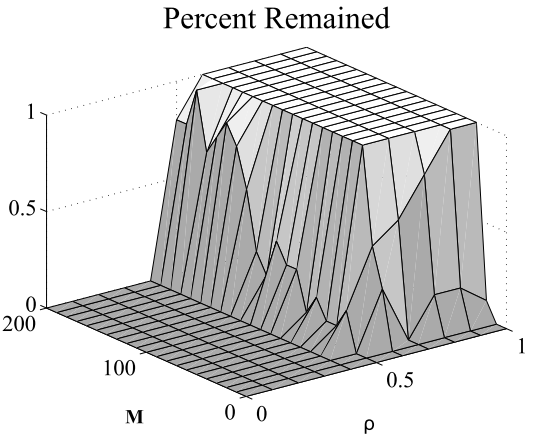
(a) Stakeholder switch from flock to torus



(b) Stakeholder switch from flock to torus



(c) Quorum sensing switch from flock to torus



(d) Quorum sensing switch from flock to torus

Figure 9.4: Switch from a flock to torus using the stakeholder model and the quorum sensing model. Stakeholders are led by attraction. Results are shown for N=400 agents.

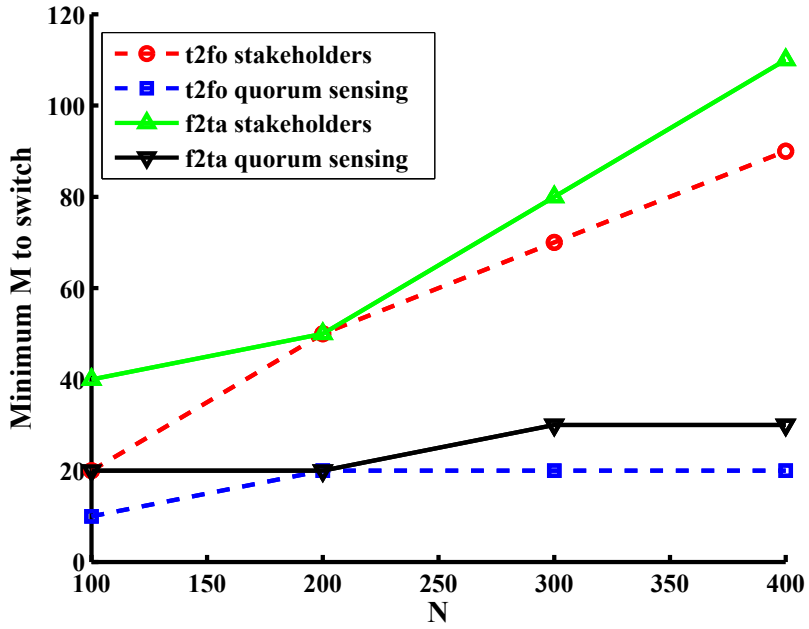


Figure 9.5: Minimum number of stakeholders, M, needed to switch from torus to flock where stakeholders are led by orientation (t2fo) and to switch from flock to torus where stakeholders are led by attraction (f2ta). Results compare the stakeholder model with the quorum sensing model.

We plotted the minimum number of agents needed to switch group types and have 100% of the simulations remain in the new group type for $N = 100$, 200, 300, and 400. Figures 9.5 and 9.6 show the minimum number of human-influenced agents, M, required to have the swarm switch group types and remain switched and the corresponding minimum percentage of total group size needed to switch group types and remain switched, respectively. Results are shown for switching from a torus to a flock where M of the agents are led by orientation (t2fo) and for switching from a flock to a torus where M of the agents are led by attraction (f2ta). We see group type switching using the quorum sensing model scales much better to higher group sizes. As the size of the group, N, increases the number of agents that the human needs to interact with stays relatively constant for the quorum sensing model, but rapidly increases for the stakeholder model.

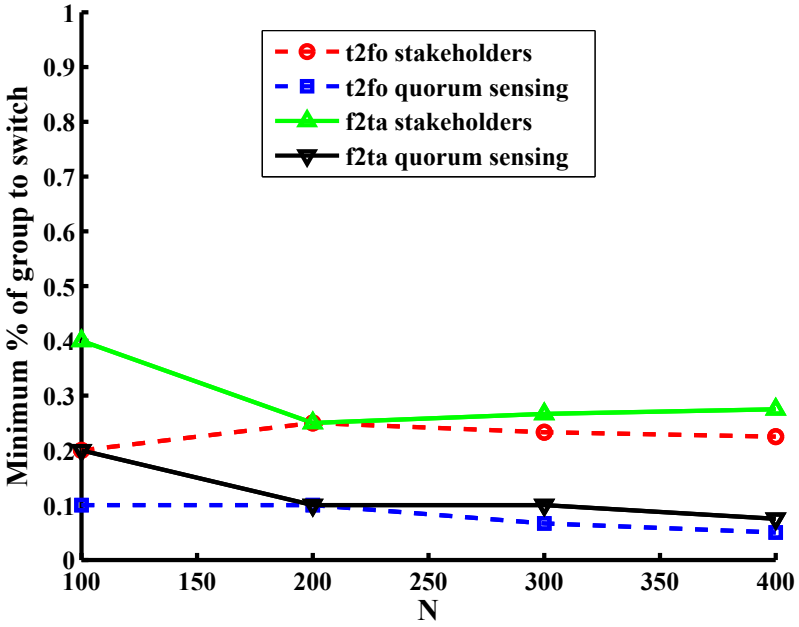


Figure 9.6: Minimum percentage of total agents that need to be stakeholders to switch from torus to flock where the stakeholders are led by orientation (t2fo) and to switch from flock to torus where the stakeholders are led by attraction (f2ta). Results compare the stakeholder model with the quorum sensing model.

9.5 Limited vulnerability

We now investigate whether using a quorum threshold allows us to limit the vulnerability of the swarm. We define limited vulnerability as the requirement that a swarm has an upper bound on the number of agents that can fail and not adversely affect the collective behavior. There is a trade-off between the responsiveness and the vulnerability of a robot swarm. A swarm that is highly responsive to changes in behavior made by only a few agents implies that if an adversary compromises a small percentage of the agents, or if a small percentage of the agents fail, then the swarm will be vulnerable to performing unwanted behaviors. This section demonstrates that quorum sensing provides a way to balance the responsiveness and vulnerability of a swarm to agent failure.

To demonstrate that quorum sensing provides a way to limit the vulnerability of a swarm, we ran a series of experiments with a nonzero quorum threshold Q where M

human-influenced stakeholders attempt to cause the group type to switch. We experimented with values of Q between 0 and 6 and values of M between 0 and 50. Because of the large number of parameters in our model we restricted our analysis to parameter settings that reliably allow the collective to switch between group types and remain switched. We used $\rho = 0.7$ when switching from a flock to a torus and $\rho = 0.5$ when switching from a torus to a flock. Based on our earlier results we chose to lead the agents by orientation when switching to a flock and to lead the agents by attraction when switching to a torus. We ran 10 replicates of each M and Q combination and used $\alpha_{max} = 1$ and $\alpha_{min} = 0$. We simulated each replicate once with the quorum sensing model and once with the stakeholder model.

We chose to compare the two models using interaction time, IT. In this case, IT is the time required to cause the collective to switch from one group behavior to another. Because we are interested in the time it takes to fully switch into a different group type we define IT as the first time that the group polarization and angular momentum are above or below a required threshold and stay above or below that threshold for the remainder of the simulation. Specifically, when switching from torus to flock

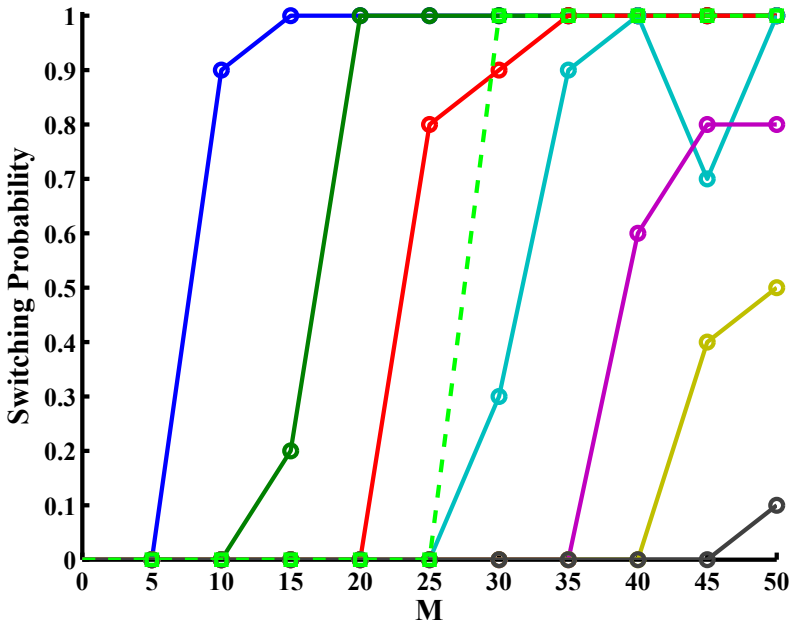
$$IT_{t2f} = \min\{t' : \forall t > t',\ p_{group}(t) > 0.75 \text{ and } m_{group}(t) < 0.25\} \tag{9.11}$$

and when switching from a flock to a torus

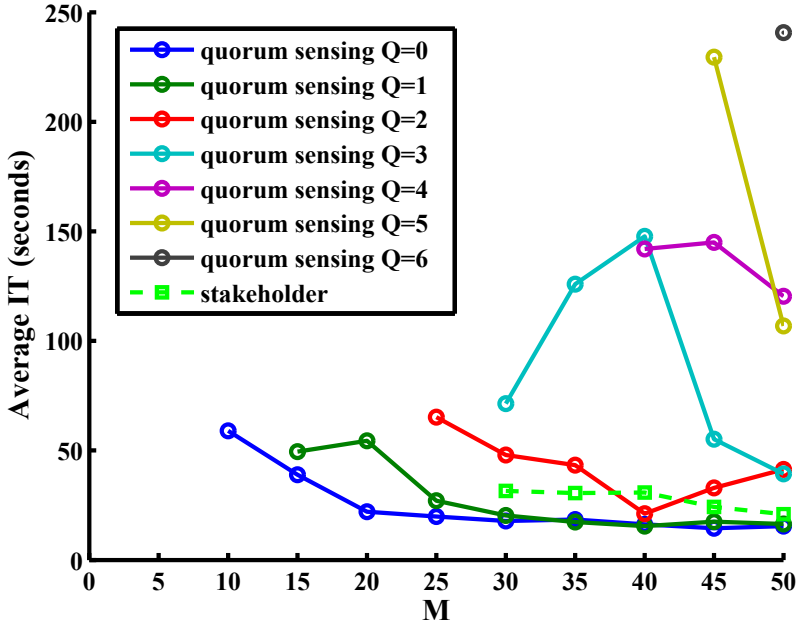
$$IT_{f2t} = \min\{t' : \forall t > t',\ p_{group}(t) < 0.25 \text{ and } m_{group}(t) > 0.75\}. \tag{9.12}$$

9.5.1 Switching from torus to flock

We plotted the probability of the swarm switching from a torus to a flock and the average IT for the trials that did switch in Figure 9.7. Figure 9.7(a) shows that the probability of switching from a torus to a flock decreases as Q increases. If we consider the stakeholders as malfunctioning or adversarial agents, then these results show that the quorum threshold Q



(a) Probability of switching from torus to flock



(b) Average interaction time of simulations that switched from a torus to a flock and remained a flock when the human influence was released

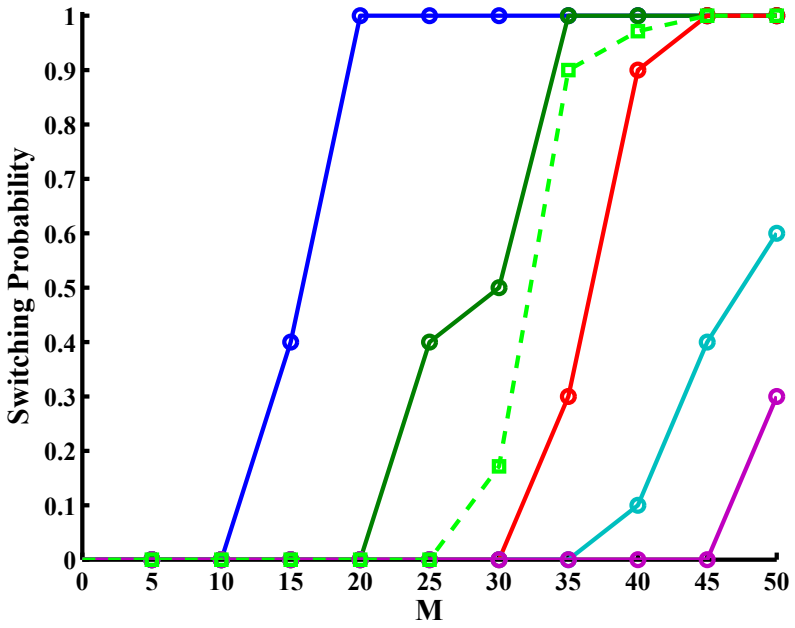
Figure 9.7: Switch from a torus to a flock using only stakeholders and using quorum sensing with Q=0–6. The M stakeholders are led by orientation. N = 100.

provides a tunable parameter that controls the vulnerability/responsiveness of the swarm. By choosing a value for Q that is sufficiently high, we can limit the responsiveness of the swarm and prevent the swarm from switching group types unless there are sufficient agents making the change. Alternatively, if the goal is to have a swarm that is highly responsive, a low quorum threshold provides controlled switches with fewer agents and with lower interaction time than are required for the stakeholder model. Thus, the quorum threshold Q provides mechanism for changing the responsiveness and vulnerability of the swarm depending on the nature of the desired swarm application. For time-sensitive or low bandwidth tasks, a low quorum threshold provides a way to quickly change the collective behavior of a swarm with limited interaction effort. For dangerous or uncertain environments, a high quorum threshold limits the vulnerability of the swarm to agent failure.

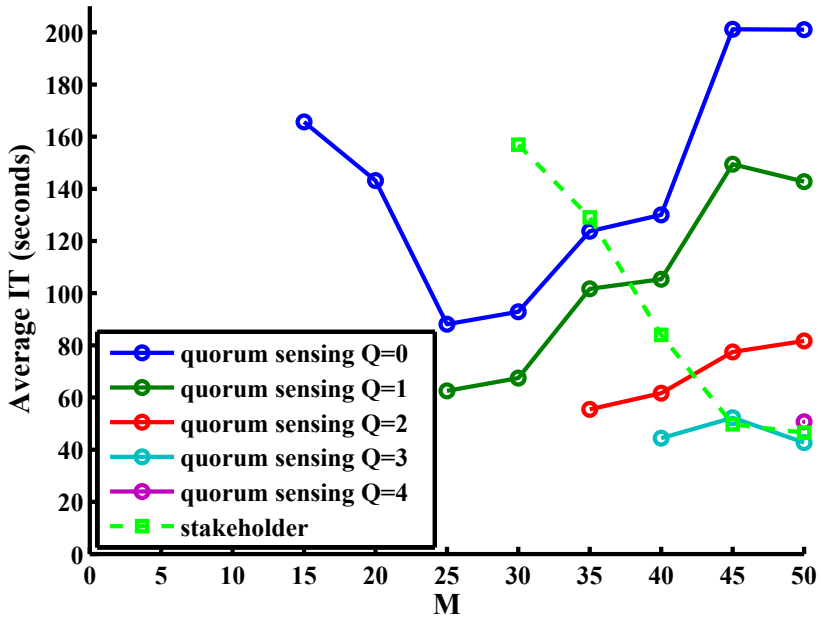
9.5.2 Switching from flock to torus

The probability of the swarm switching from a flock to a torus and the average IT for the trials that did switch are shown in Figure 9.7. Figure 9.8(b) shows that the quorum sensing model can switch from a flock to a torus group type for smaller values of M than the stakeholder model. Additionally, we see that changing the quorum threshold Q allows us to limit the vulnerability of the swarm by increasing or decreasing the probability that the swarm will be responsive to a certain number of agents changing their behavior. Interestingly, when $M = 45$ or 50, the stakeholder model actually has a smaller average interaction time than the quorum sensing model. Additionally, as Q increases, the interaction time decreases for large enough M. This is a reversal of the trend seen in Figures 9.7(b) for switching from a torus to a flock.

We investigated this interesting phenomenon and found that it resulted from using $\alpha^{max} = 1$ combined with a low Q. This is because when $\alpha_{max} = 1$, type-aware agents will ignore all of their neighbors except for neighbors that are human-aware. When Q is low and M is high, agents will almost always have at least one human-aware stakeholder in their



(a) Probability of switching from flock to torus



(b) Average interaction time of simulations that switched from a flock to a torus and remained a torus when the human influence was released

Figure 9.8: Switch from a flock to a torus using only stakeholders and using quorum sensing with Q = 0–4. The M stakeholders are led by attraction. N=100.

neighborhood and so will ignore the rest of their neighbors. This results in the type-aware agents ignoring each other which causes the agents to form clumps around the torus perimeter which raises p_{group} above 0.25 and prevents the torus from fully forming until the human influence is released. On the other hand, because the stakeholder model has no quorum sensing or type-awareness, the agents space nicely around the torus and do not form clumps resulting in a stable p_{group} value less than 0.25 early on in the simulation thus resulting in a lower interaction time.

This can be seen visually in Figure 9.9 where we have plotted the p_{group} resulting from a single simulation of the stakeholder model and a single simulation the quorum sensing model for the task of switching from a flock to a torus. Figure 9.9 shows that while the quorum sensing model provides a smoother change, the p_{group} never drops low enough to qualify as a torus until after the human influence is released ($t = 225$). This also explains why increasing Q decreases IT. Increasing Q makes it less likely for type-aware agents to increase their type-awareness and therefore less likely to ignore many of their neighbors and form clumps. The reader is referred to Appendix C.2 for an empirical analysis on the effects of changing α^{max}.

9.6 Summary

We have shown that quorum sensing increases the scalability of human-swarm interactions. We have additionally shown that the quorum threshold Q provides mechanism for limiting the vulnerability of a swarm. The majority of the analysis in this chapter has focused on homogeneous parameters for quorum sensing. We also experimented with heterogeneous values of Q_i, α_i^{max}, and α_i^{min}, where the parameter value for agent i is drawn from a probability distribution. We found that the quorum sensing model was robust to this heterogeneity and that adding parameter heterogeneity actually slightly decreases the minimum number of stakeholders needed to allow reliable switching between group types (see Appendix C.3).

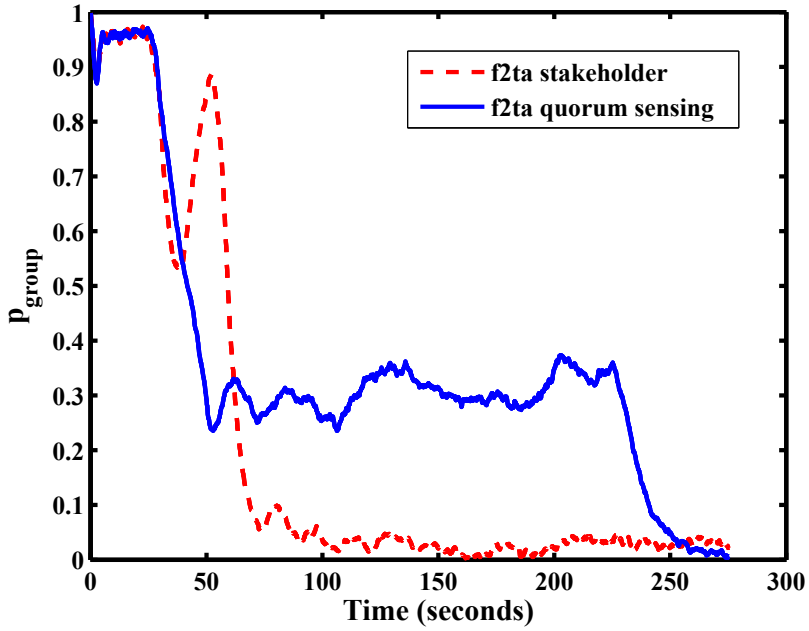


Figure 9.9: Group polarization over time for one simulation of switching between flock to torus using the stakeholder model and using the quorum sensing model. Human influence is applied at $t = 25$ and released at $t = 225$.

Chapter 10

Conclusions and Future Work

This thesis focused on two major problems that impede scalable human interaction with large bio-inspired robot swarms. First, how can we accurately detect and estimate the collective state of a robot swarm using limited information from a subset of the swarm? Second, how can we increase the scalability of human influence over the collective behavior of a swarm while limiting the vulnerability of the swarm to agent failures?

We have presented a bio-inspired model of swarming that requires no explicit communication between agents and exhibits two fundamental attractors: a flock and a torus. Using this model, we showed that even if agents are not capable of determining their location or heading, we can accurately classify the group behavior of the swarm using local samples from individual agents. We showed that sampling the number of neighbors and the angular velocity of only 5% of the swarm is sufficient to achieve 100% classification accuracy on a test set of 100 simulations of our model. We additionally showed the existence of neglect benevolence [37] when classifying the collective behavior of our model and provided evidence that our method of detecting group type from limited samples will scale to larger swarm sizes. When agents are capable of localization, we demonstrated that low error estimates of the actual group centroid and group heading can be found using a limited number of samples of local information from agents in the group.

To better evaluate human-swarm interactions, we introduced two metrics, observation effort and estimation effort, and used them to compare different classification and estimation techniques. We also considered two different swarm models and showed how our methods

for group type classification generalize to other multi-agent systems that exhibit distinct collective behaviors. We hypothesize that this methodology of estimating global properties from local samples is applicable to many other systems where global properties emerge from individual agent interactions.

Finally, we addressed the problem of increasing the scalability of human-swarm interactions while limiting the swarm's vulnerability to agent failures. We demonstrated how to incorporate a bio-inspired form of quorum sensing into an existing swarm model. We then compared human-influenced group type switching using our quorum sensing model and the previously developed stakeholder model [36]. We showed that quorum sensing increases the scalability of human-swarm interactions while also providing a mechanism for limiting the vulnerability of the swarm to agent failure through a quorum threshold. Thus, the quorum threshold provides a tunable parameter that allows a human to choose the desired responsiveness, and corresponding vulnerability, of a swarm depending on the application.

10.1 Future work

Future work should investigate how well the group type classification methods described in this thesis scale to larger swarm sizes and also investigate how human or environmental influences affect classification accuracy. Group type classification and collective state estimation should also be applied to additional swarm models that exhibit emergent behaviors.

We have derived a theoretical result for the maximum speed of a moving torus (see Appendix D). Future work should examine how this result could be incorporated into group type classification. We have also performed a preliminary analysis of the discrete-time approximation error in an attraction-based version of our swarm model (see Appendix E). A more thorough analysis for our full model should be performed to determine the sensitivity to simulation step size and other model parameters.

Future work should also include designing algorithms that allow agents to change their quorum thresholds over time, and performing user studies with simulated or actual robots.

These user studies should investigate how the ideas of observation effort, estimation effort, and quorum sensing directly impact human-swarm interactions and should also investigate potential applications for swarms with multiple behaviors.

One potential application for the work described in this thesis is the area of plume tracing, where a swarm of agents attempt to locate the source of a chemical leak. If agents form a flock as they collectively follow the chemical gradient and form a torus around the source, then our group type classification and estimation techniques would allow us to detect when the swarm has found a potential leak and estimate the location of the leak.

Appendix A

Classification using number of neighbors and absolute angular velocity

The classification accuracies and observation effort for our swarm model when using number of neighbors and absolute angular velocity are shown in Figures A.1, A.2, and A.3. The results were nearly identical to classification using only number of neighbors.

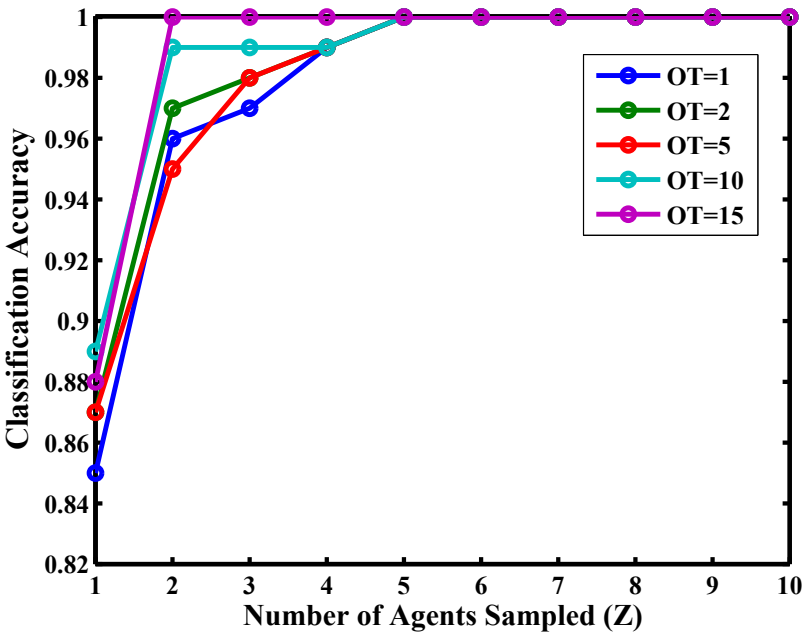


Figure A.1: Probability of group type detection as the number of agents sampled, Z, increases for different observation times, OT. For N = 100 agents.

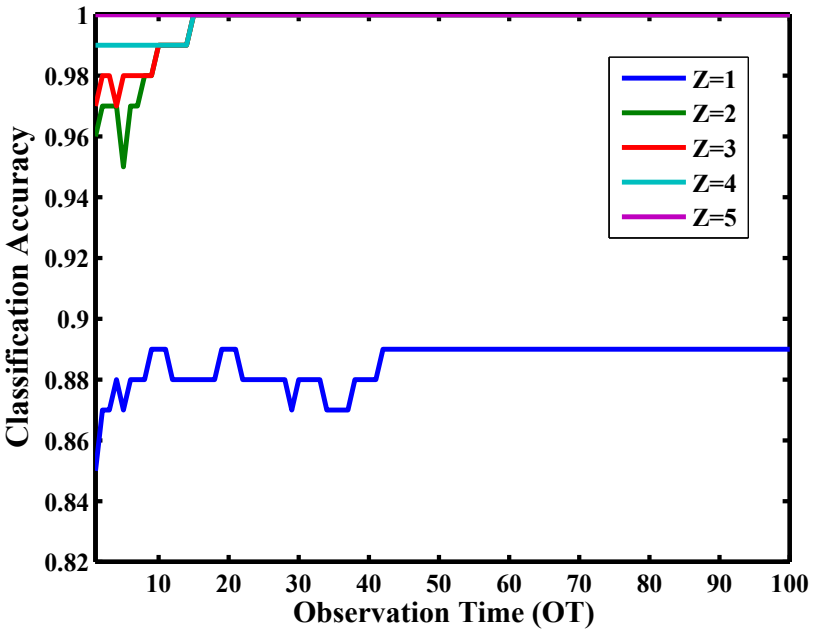


Figure A.2: Probability of group type detection as observation time, OT, increases for different numbers of agents sampled, Z. For N = 100 agents.

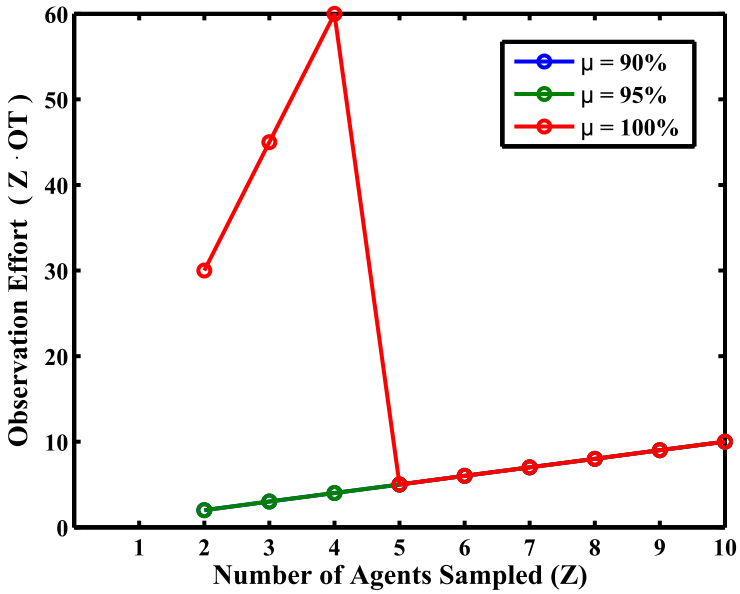


Figure A.3: Observation effort over Z and for different accuracy thrsholds μ. Using $Z = 1$ never resulted in the desired accuracy so the observation effort is infinity. The results for $\mu = 90\%$ aand $\mu = 95\%$ were identical. Results are for N = 100 agents.

Appendix B

Results for classification error analysis

B.1 Error analysis for classification using only the number of neighbors

For $Z = 1$ and $OT = 1$ the confusion matrix is given in Table B.1. These results show that using only the number of neighbors has a classification bias toward predicting torus as the swarm group type when only one sample is used but still achieves a remarkably high accuracy of 82% accuracy. If we add an additional sample from a different agent ($Z = 2$), the resulting confusion matrix is shown in Table B.2. This provides a large improvement by decreasing the error by 66.67% and raising the total accuracy to 94%.

If, instead of increasing Z, we keep $Z = 1$ and sample twice from the same agent, we get the confusion matrix shown in Table B.3. This result shows that additional samples corrected the misclassifications of a torus as a flock, but only slightly improved the misclassification of a flock as a torus. The error was decreased by 27.78% and the total accuracy improved to 87%. When we compare these results to the results for $Z = 2$ and $OT = 1$ we see, as noted earlier, that additional samples from a single agent do not help as much as additional

		Estimated Class		
		Flock	Torus	Total
Actual Class	Flock	32	15	47
	Torus	3	50	53
	Total	35	65	100

Table B.1: Confusion matrix for $Z = 1$ and $OT = 1$ using only the number of neighbors as a feature.

		Estimated Class		
		Flock	Torus	Total
Actual Class	Flock	42	5	47
	Torus	1	52	53
	Total	43	57	100

Table B.2: Confusion matrix for $Z = 2$ and $OT = 1$ using only the number of neighbors as a feature.

		Estimated Class		
		Flock	Torus	Total
Actual Class	Flock	34	13	47
	Torus	0	53	53
	Total	34	66	100

Table B.3: Confusion matrix for $Z = 1$ and $OT = 2$ using only the number of neighbors as a feature.

samples from a different agent. Additionally it is interesting to note that values of $OT > 2$ did not provide more than 1% improvement over the accuracy obtained when $OT = 2$.

Examining the simulations from the test set that were incorrectly classified, we found that the main reason that flocks are often classified as a torus for $Z = 1$ turns out to be the long thin tail that often forms behind a flock when agents start from random initial conditions. As shown in Figure 6.7, when sampled agents are in the tail they have fewer neighbors and our detection algorithm predicts torus as the most likely group type. Another reason is that the agent sampled may be on the edge of the swarm (see Figure B.1) which also can result in a low number of neighbors. It should be noted, however, that given enough time the flock will fully clump together eliminating the long tail and forming a flock similar to that shown in Figure 2.1(c). However, these results show that our classification method performs well even when the flock group type has not fully formed.

We also investigated the torus formations that were misclassified as flocks. Figure B.2 shows one misclassified example. Notice that in this case the group type has fully formed, but because of the inherent noise caused by our stochastic topology, agents occasionally have a much larger (or smaller) number of neighbors than is typical for a certain group type. In

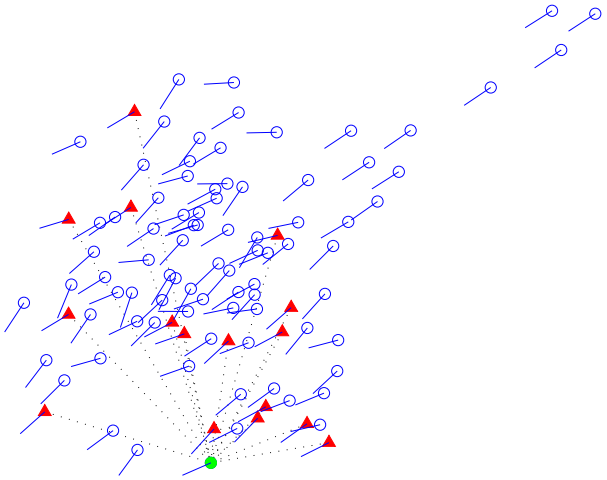


Figure B.1: An agent sampled (the green disc) from the edge of the flock has fewer neighbors (the 16 red triangles) than those in the main clump of the flock resulting in the misclassification of the flock as a torus.

		Predicted		
		Flock	Torus	Total
Actual	Flock	29	18	47
	Torus	13	40	53
	Total	42	58	100

Table B.4: Confusion matrix for $Z = 1$ and $OT = 1$ using only absolute angular velocity as a feature.

this instance, the sampled agent had 17 neighbors at the time of sampling which is more likely for a flock than for a torus (see Figure 6.3) resulting in a misclassification.

B.2 Error analysis for classificiation using only absolute angular velocity

The confusion matrix for $Z = 1$ and $OT = 1$ is shown in Table B.4. We see that there are approximately equal misclassifications. If we set Z = 2 and OT = 1, (see Table B.5) we have a slight improvement from 69% to 72% accuracy, but, while the number of misclassifications of a torus as a flock decreased, there was a slight increase in the number of flocks misclassified

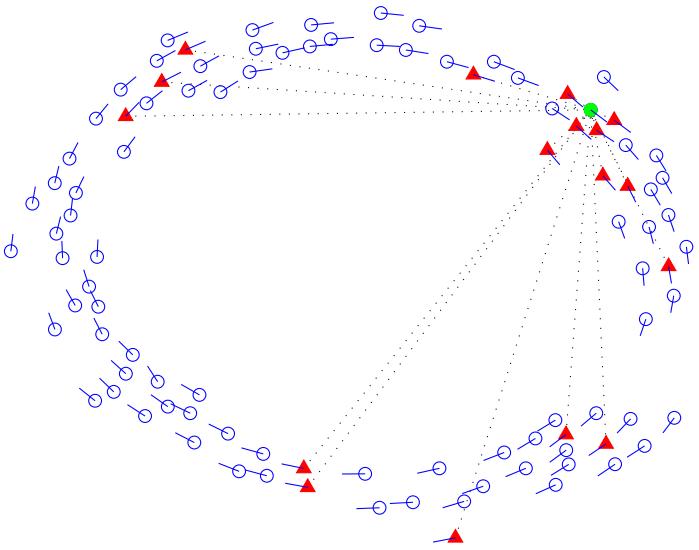


Figure B.2: Torus that was misclassified as a flock. The agent has 17 neighbors. Figure 6.3 shows that the probability of an agent having 17 neighbors is more likely for a flock than for a torus resulting in a misclassification of the torus using one sample from one agent.

		Predicted		
		Flock	Torus	Total
Actual	Flock	28	19	47
	Torus	9	44	53
	Total	37	63	100

Table B.5: Confusion matrix for $Z = 2$ and $OT = 1$ using only absolute angular velocity as a feature.

		Predicted		
		Flock	Torus	Total
Actual	Flock	29	18	47
	Torus	12	41	53
	Total	41	59	100

Table B.6: Confusion matrix for $Z = 1$ and $OT = 2$ using only absolute angular velocity as a feature.

		Predicted		
		Flock	Torus	Total
Actual	Flock	44	3	47
	Torus	1	52	53
	Total	45	55	100

Table B.7: Confusion matrix for $Z = 5$ and $OT = 5$ using only absolute angular velocity as a feature.

as a torus. If we sample twice with the same agent, (see Table B.6) we only have a slight improvement over a single sample from a single agent. However, Table B.7 shows that increasing Z to 5 and OT to 5 dramatically improves accuracy to 96%.

B.3 Error analysis for classification using angular velocity

In the following confusion matrices we abbreviate clockwise as (cw) and counterclockwise as (ccw). The confusion matrix for Z=1 and OT =1 is shown in Table B.8. If we let Z = 2 and OT = 1 (see Table B.9) there is a large improvement from 71% to 79% accuracy. If we let $OT = 2$ and $Z = 1$ (see Table B.10) there is only a slight increase in accuracy compared to

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	28	13	6	47
	CCW Torus	6	21	0	27
	CW Torus	4	0	22	26
	Total	38	34	28	100

Table B.8: Confusion matrix for $Z = 1$ and $OT = 1$ using only angular velocity to classify group type and rotation.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	30	9	8	47
	CCW Torus	1	26	0	27
	CW Torus	3	0	23	26
	Total	34	25	31	100

Table B.9: Confusion matrix for $Z = 2$ and $OT = 1$ using only angular velocity to classify group type and rotation.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	27	12	8	47
	CCW Torus	3	24	0	27
	CW Torus	4	0	22	26
	Total	26	39	35	100

Table B.10: Confusion matrix for $Z = 1$ and $OT = 2$ using only angular velocity to classify group type and rotation.

sampling one agent for one time step. Finally, increasing Z to 5 and OT to 5 we reach 99% accuracy over the test set with just one flock misclassified as a torus as shown in Table B.11.

B.4 Error analysis for classification during transient behaviors

The confusion matrix for $Z = 1$ at time 10 is shown in Table B.12. The confusion matrix for $Z = 5$ at simulation time 30 is shown in Table B.13

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	46	1	0	47
	CCW Torus	0	27	0	27
	CW Torus	0	0	26	26
	Total	19	41	40	100

Table B.11: Confusion matrix for $Z = 5$ and $OT = 5$ using only angular velocity to classify group type and rotation.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	8	17	16	41
	CCW Torus	1	10	16	27
	CW Torus	2	18	12	32
	Total	11	45	44	100

Table B.12: Confusion matrix for group type classification when sampling angular velocity and number of neighbors with $Z = 1$ and $OT = 1$ after 10 seconds of simulation.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	30	4	7	41
	CCW Torus	0	24	3	27
	CW Torus	0	6	26	32
	Total	30	34	36	100

Table B.13: Confusion matrix for group type classification when sampling angular velocity and number of neighbors with $Z = 5$ and $OT = 1$ after 30 seconds of simulation.

B.5 Error analysis for two-group classification in Couzin's model

When $Z = 1$ and $OT = 1$ we have the confusion matrix shown in Table B.14. The confusion matrix for $Z = 5$ and $OT = 5$ is shown in Table B.15. Table B.16 shows the confusion matrix for $Z = 15$ and $OT = 15$.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	5	21	24	50
	CCW Torus	6	13	4	23
	CW Torus	6	6	12	24
	Total	17	40	40	97

Table B.14: Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 1$ and $OT = 1$.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	29	10	11	50
	CCW Torus	1	22	0	23
	CW Torus	4	0	20	24
	Total	34	31	32	97

Table B.15: Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 5$ and $OT = 5$.

		Predicted			
		Flock	CCW Torus	CW Torus	Total
Actual	Flock	45	3	2	50
	CCW Torus	2	21	0	23
	CW Torus	3	0	21	24
	Total	34	31	32	97

Table B.16: Confusion matrix for classifying group types in Couzin's model with $R_o = 6$ sampling angular velocity and number of neighbors for $Z = 15$ and $OT = 15$.

B.6 Error analysis for three-group classification for Couzin's model

The confusion matrix for $Z = 1$ and $OT = 1$ is shown in Table B.17. For both the flock and swarm we see that the misclassifications outnumber the accurate classifications. The misclassifications are lower for the clockwise and counterclockwise torus, but are still very high. Thus, one sample from a single agent tells very little about the group type with an overall accuracy on the test set of just 31.53%. However, if we let $Z = 5$ and $OT = 5$ (see Table B.18) the overall accuracy jumps to 55.59%. If we let $Z = 15$ and $OT = 15$ (see Table B.19) the accuracy rises to 85.42% overall and we see that all but one misclassification results from misclassifying a group type as a swarm (fourth column) or misclassifying a swarm as another group type (fourth row).

		Predicted				
		Flock	CCW Torus	CW Torus	Swarm	Total
Actual	Flock	19	31	35	17	102
	CCW Torus	4	30	11	5	50
	CW Torus	2	11	28	2	43
	Swarm	8	39	37	16	100
	Total	33	111	111	40	295

Table B.17: Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 1$ and $OT = 1$.

		Predicted				
		Flock	CCW Torus	CW Torus	Swarm	Total
Actual	Flock	49	16	8	29	102
	CCW Torus	3	41	0	6	50
	CW Torus	2	3	30	8	43
	Swarm	11	16	29	44	100
	Total	65	76	67	87	295

Table B.18: Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 5$ and $OT = 5$.

		Predicted				
		Flock	CCW Torus	CW Torus	Swarm	Total
Actual	Flock	83	1	0	18	102
	CCW Torus	0	49	0	1	50
	CW Torus	0	0	39	4	43
	Swarm	9	6	4	81	100
	Total	92	56	43	104	295

Table B.19: Confusion matrix for classifying all four group types in Couzin's model, sampling angular velocity and number of neighbors for $Z = 15$ and $OT = 15$.

B.7 Tables of classifier accuracies

For comparison between different classifiers discussed in Chapter 6, we have included the accuracies that resulted from using $Z \in [1, 2, \ldots, 10]$ and $OT \in [1, 2, \ldots, 10]$ in Tables B.20, B.21, B.22, and B.23. The results for group type classification using Couzin's model are shown in Table B.24.

		Observation Time (OT)									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Z	**1**	82	87	86	87	87	87	87	87	88	88
	2	94	96	96	95	95	96	96	95	97	98
	3	97	97	97	97	97	97	98	98	98	99
	4	98	99	99	99	99	99	99	99	99	99
	5	100	100	100	100	100	100	100	100	100	100
	6	100	100	100	100	100	100	100	100	100	100
	7	100	100	100	100	100	100	100	100	100	100
	8	100	100	100	100	100	100	100	100	100	100
	9	100	100	100	100	100	100	100	100	100	100
	10	100	100	100	100	100	100	100	100	100	100

Table B.20: Percent accuracy for using only number of neighbors to predict the unoriented group type of a swarm.

		Observation Time (OT)									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Z	**1**	69	70	73	72	75	74	81	79	79	81
	2	72	77	80	80	79	87	88	90	91	90
	3	82	84	91	94	91	92	96	98	98	97
	4	89	90	89	92	95	95	96	97	98	98
	5	89	93	93	94	96	98	98	98	98	97
	6	93	94	94	97	97	97	97	97	97	98
	7	94	95	95	97	98	98	99	99	98	98
	8	95	97	98	99	100	100	100	100	100	100
	9	95	98	100	100	100	100	100	100	100	100
	10	96	99	100	100	100	100	100	100	100	100

Table B.21: Percent accuracy for using only angular velocity to predict the unoriented group type of a swarm.

		Observation Time									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Z	**1**	71	73	78	78	82	85	85	83	85	90
	2	79	80	84	87	88	93	95	97	97	97
	3	90	91	92	95	97	97	98	99	99	99
	4	96	95	95	97	96	99	99	100	100	100
	5	95	96	98	98	99	99	99	99	99	100
	6	97	96	98	99	100	100	100	100	100	100
	7	99	98	99	99	99	99	100	100	100	100
	8	98	100	100	100	100	100	100	100	100	100
	9	98	100	100	100	100	100	100	100	100	100
	10	99	100	100	100	100	100	100	100	100	100

Table B.22: Percent accuracy for using only angular velocity to predict the group type and, in the case of the torus, orientation of the swarm.

		Observation Time									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Z	**1**	84	85	86	86	87	88	88	88	88	89
	2	95	96	97	97	95	97	98	97	98	99
	3	97	98	98	97	98	98	98	98	98	100
	4	99	99	99	99	99	99	99	99	99	99
	5	100	100	100	100	100	100	100	100	100	100
	6	100	100	100	100	100	100	100	100	100	100
	7	100	100	100	100	100	100	100	100	100	100
	8	100	100	100	100	100	100	100	100	100	100
	9	100	100	100	100	100	100	100	100	100	100
	10	100	100	100	100	100	100	100	100	100	100

Table B.23: Percent accuracy for using both angular velocity and number of neighbors to predict the group type and, in the case of the torus, orientation of the swarm.

		Observation Time (OT)									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Z	**1**	30.93	36.08	38.14	36.08	38.14	44.33	43.30	39.18	36.08	39.18
	2	45.36	40.21	43.30	46.39	44.33	51.55	51.55	53.61	56.70	56.70
	3	48.45	52.58	59.79	56.70	58.76	60.82	62.89	65.98	62.89	61.86
	4	56.70	59.79	60.82	61.86	63.92	67.01	70.10	72.16	73.20	73.20
	5	56.70	60.82	65.98	72.16	73.20	72.16	72.16	73.20	73.20	73.20
	6	55.67	60.82	62.89	68.04	69.07	70.10	68.04	70.10	72.16	71.13
	7	58.76	65.98	70.10	69.07	69.07	68.04	73.20	73.20	74.23	75.26
	8	63.92	70.10	74.23	74.23	75.26	76.29	75.26	74.23	76.29	74.23
	9	58.76	68.04	73.20	76.29	75.26	76.29	74.23	74.23	73.20	72.16
	10	68.04	72.16	76.29	76.29	81.44	78.35	80.41	80.41	78.35	77.32

Table B.24: Percent accuracy for using only number of neighbors to predict the oriented group type for Couzin's model.

Appendix C

Supplementary Results for Quorum Sensing

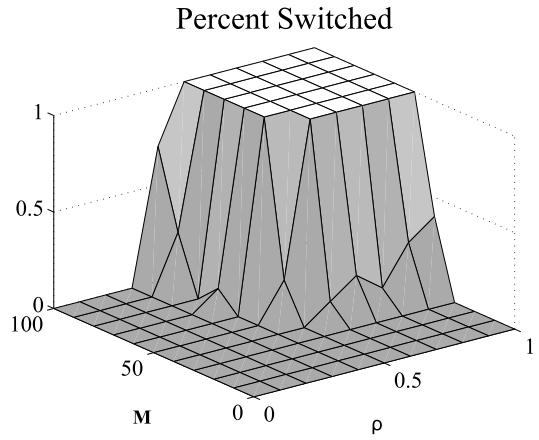
C.1 Scalability

Figures C.1–C.4 compare the scalability of the quorum sensing model with the stakeholder model for $N = 200$ and $N = 300$.

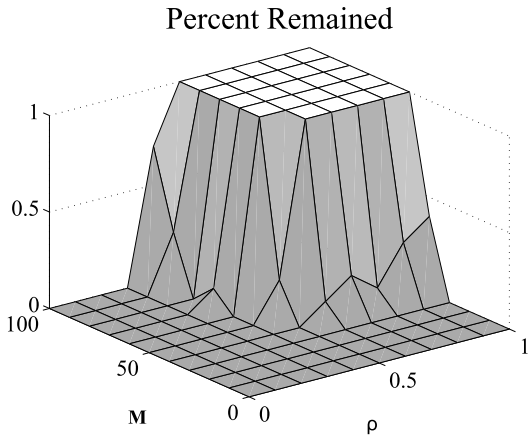
C.2 Sensitivity analysis

We ran a series of simulations for values of α^{max} between 0 and 1 with $Q = 0$ and took the average over 10 replicates. Values of α^{min} less than 0.5 did not allow reliable switching between group types so we do not report those results. For $0.5 \leq \alpha^{max} \leq 1$, the average interaction time required to switch from a flock to a torus is shown in Figure C.5. We see that for $0.5 \leq \alpha^{max} \leq 0.7$ the interaction time is less than that of the stakeholder model when $M = 45$ and $M = 50$, additionally we see that when $\alpha^{max} = 0.6$ the collective behavior can switch for $M \geq 30$ and the interaction time is minimized for these values of M. Because higher values of α^{max} have much higher interaction times and work for similar values of M, we chose to restrict the analysis in the subsequent sections to $\alpha^{max} = 0.6$.

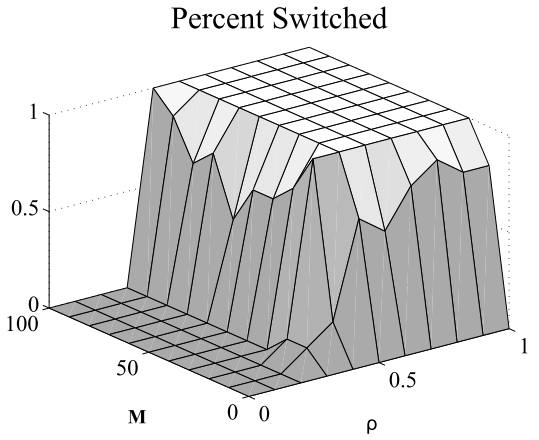
To see the effect this has on interaction time for different values of Q, we replicated Figure 9.8(b) using $\alpha^{min} = 0.6$ instead of 1. The results for switching from a flock to a torus are shown in Figure C.6. From this figure we see that using $\alpha^{max} = 0.6$ makes



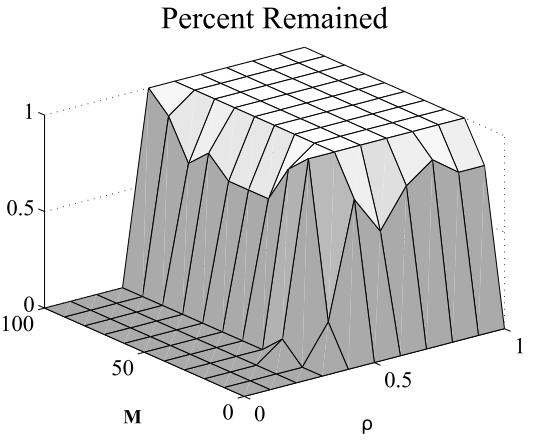
(a) Stakeholder switch from torus to flock



(b) Stakeholder switch from torus to flock

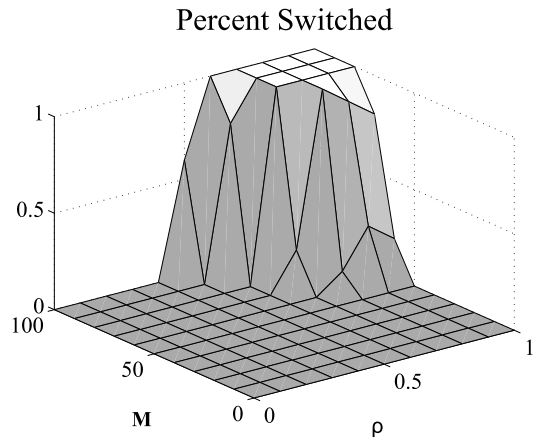


(c) Quorum sensing switch from torus to flock

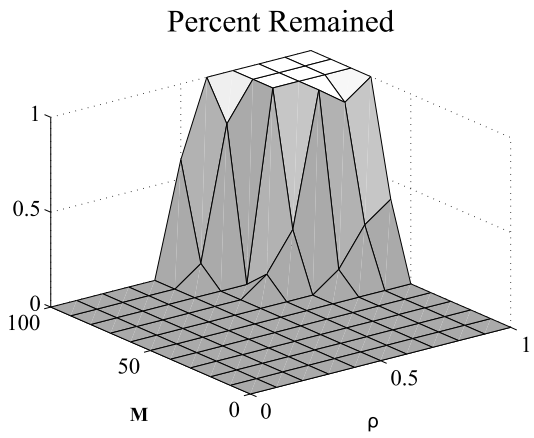


(d) Quorum sensing switch from torus to flock

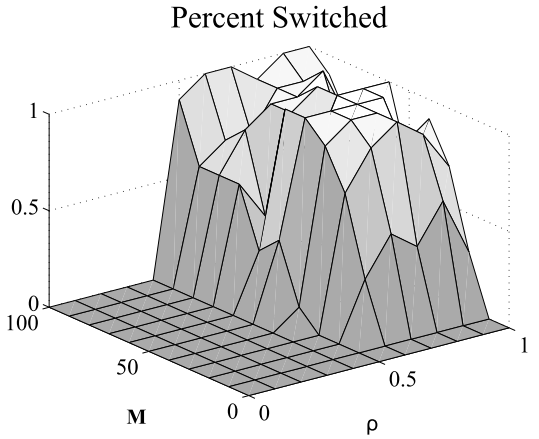
Figure C.1: Switch from a torus to a flock using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by orientation. N=200.



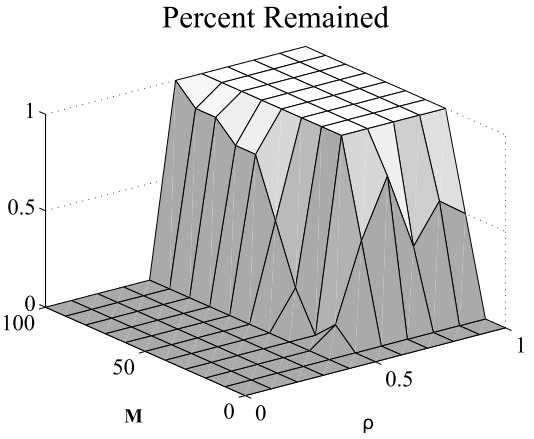
(a) Stakeholder switch from flock to torus



(b) Stakeholder switch from flock to torus

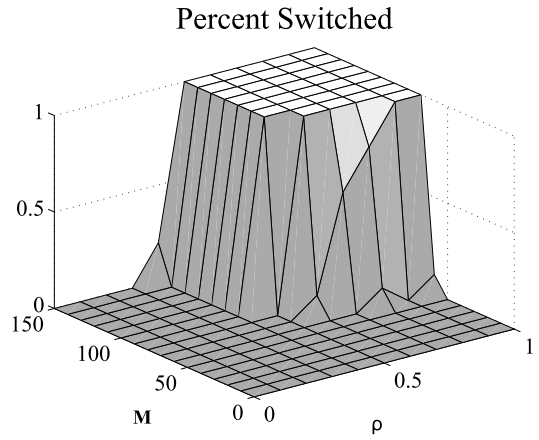


(c) Quorum sensing switch from flock to torus

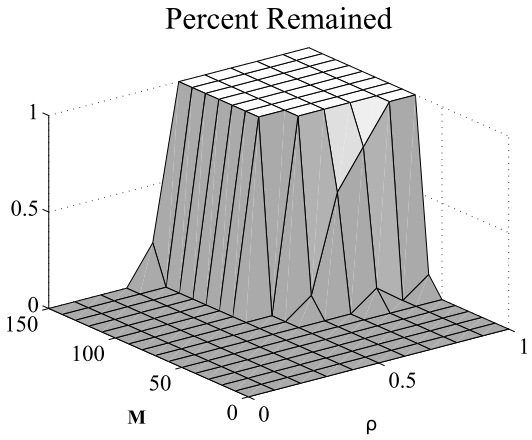


(d) Quorum sensing switch from flock to torus

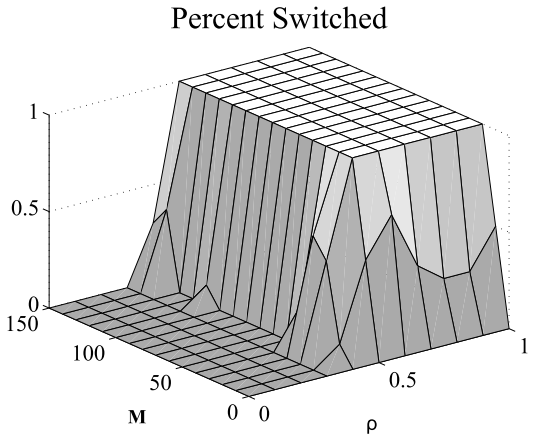
Figure C.2: Switch from a flock to torus using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by attraction. N=200.



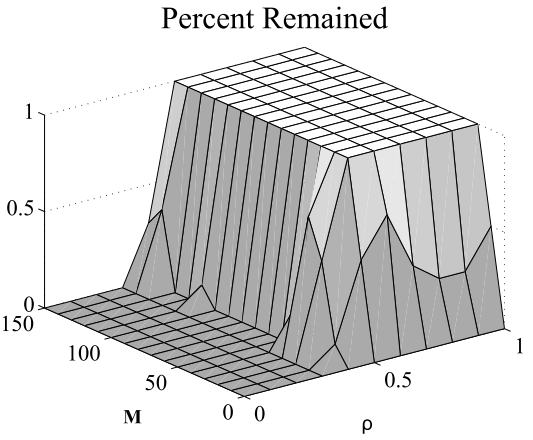
(a) Stakeholder switch from torus to flock



(b) Stakeholder switch from torus to flock

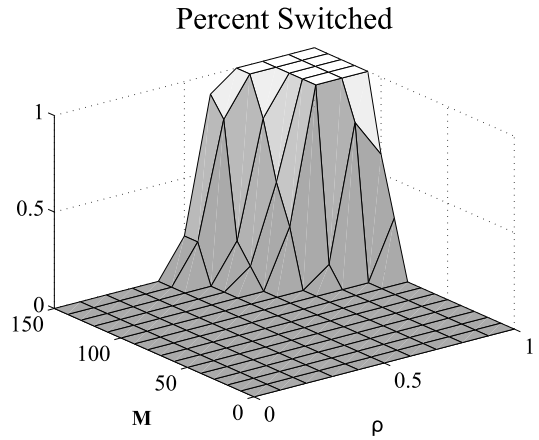


(c) Quorum sensing switch from torus to flock

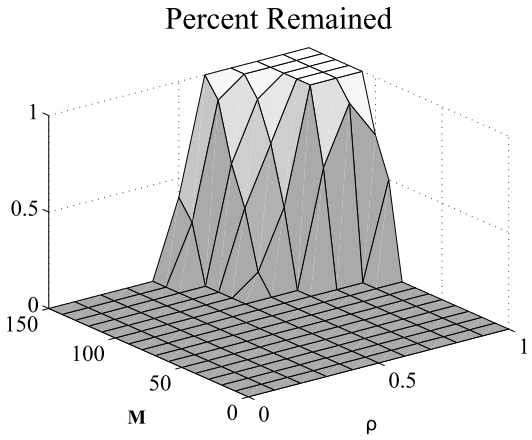


(d) Quorum sensing switch from torus to flock

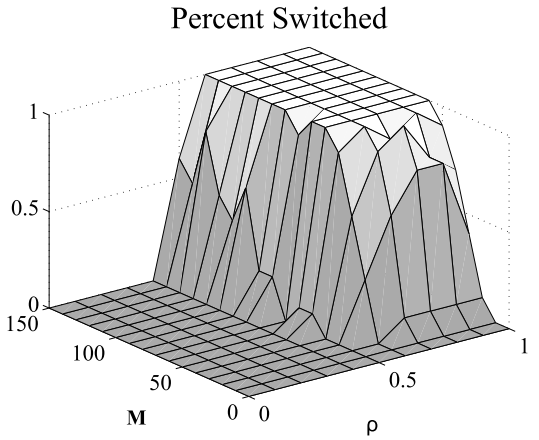
Figure C.3: Switch from a torus to a flock using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by orientation. N=300.



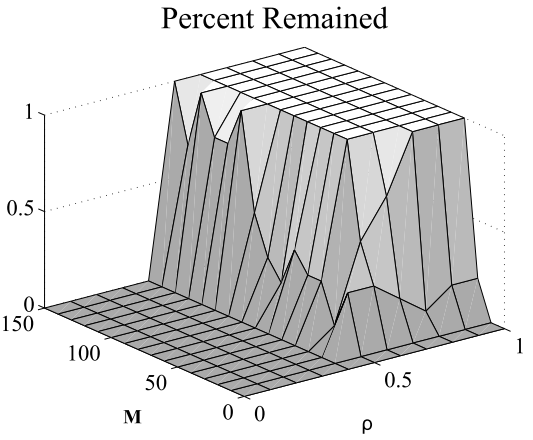
(a) Stakeholder switch from flock to torus



(b) Stakeholder switch from flock to torus



(c) Quorum sensing switch from flock to torus



(d) Quorum sensing switch from flock to torus

Figure C.4: Switch from a flock to torus using only stakeholders and using stakeholders and type aware agents. Stakeholders are led by attraction. N=300.

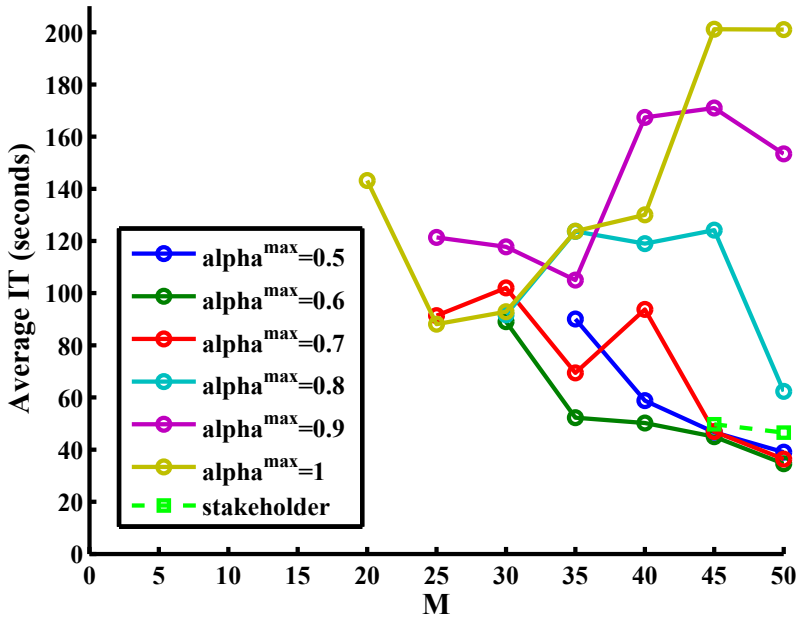


Figure C.5: Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $Q = 0$ and $0.1 \leq \alpha^{max} \leq 1$.

the interaction time for the quorum sensing model lower than the interaction time for the stakeholder model. Additionally, this figure shows that the quorum sensing model allows for the vulnerability/responsiveness of the model to be adjusted by changing Q. If $Q = 2$ the model will not reliably switch behaviors unless 50 or more agents receive human influence. Once the human controls 50 agents we see that the interaction time is about 10 seconds faster than the stakeholder model with the same number of stakeholders.

We also looked at different values of α^{max} for switching from a torus to a flock. Figure C.7 shows that α^{max} has less impact on switching from a torus to a flock than it does for switching from a flock to a torus. For every value of α^{max} shown, the interaction time is very similar with $\alpha^{max} = 0.5$ performing the worst and $\alpha^{max} = 0.8$ having the largest region of switching. There is no clear winner in terms of switching region or interaction time. In the following analysis, we chose to focus on $\alpha^{max} = 0.8$ because it allowed group type switches over the largest range of M and had reasonably low average interaction times.

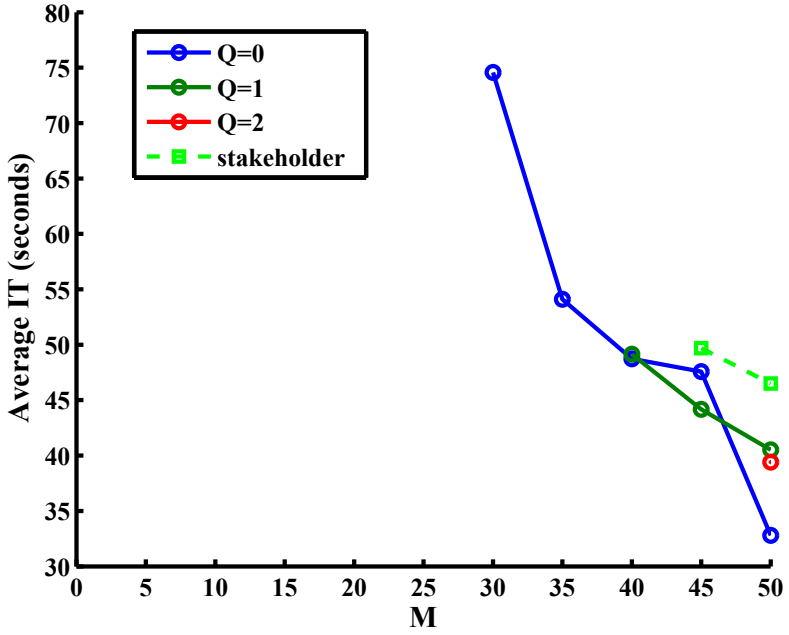


Figure C.6: Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $\rho = 0.7$, $Q = 0, 1, 2$, $\alpha^{max} = 0.6$, and $\alpha^{min} = 0$.

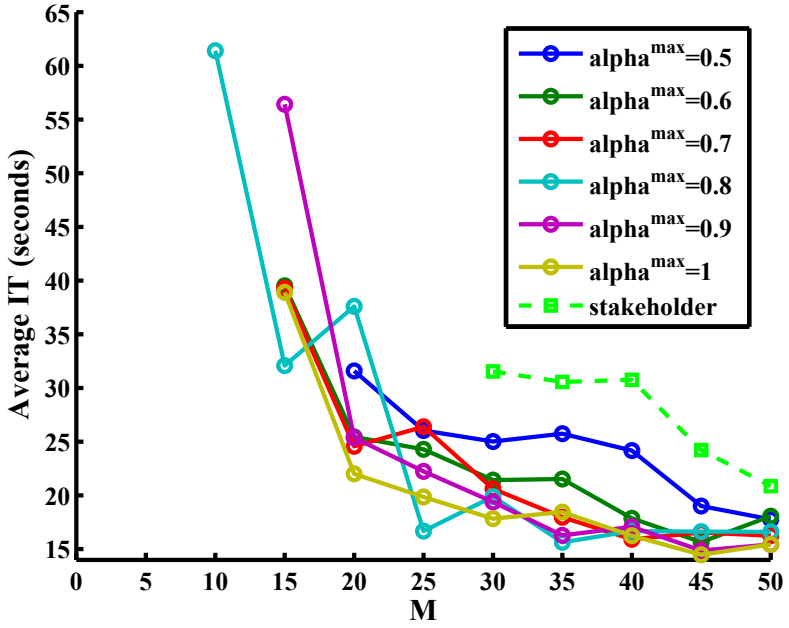


Figure C.7: Average interaction time to switch from torus to flock for stakeholder model, and quorum sensing model for $Q = 0$ and $0.1 \leq \alpha^{max} \leq 1$.

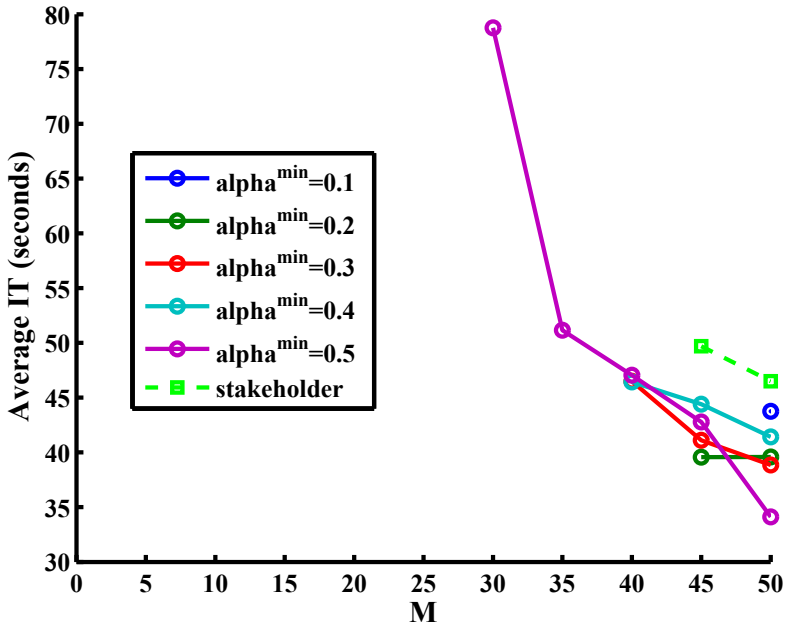


Figure C.8: Average interaction time to switch from flock to torus for stakeholder model, and quorum sensing model for $Q = 2$, $\alpha^{max} = 0.6$, and $0.1 \leq \alpha^{min} \leq 0.5$.

To study the sensitivity of the model to different values of α^{min} we ran two sets of simulations, one for switching from flock to torus and one for switching from torus to flock, and varied α^{min}. For the flock to torus simulation we used $\alpha_i^{max} = 0.6$, $Q_i = 2$, and $\alpha_i^{max} \in [0.1, 0.5]$, $\forall i$. We ran 10 replicates for each value of α^{min} in 0.1 unit increments. The results are shown in Figure C.8. We see that increasing α^{min} has the general effect of lowering interaction time as well as increasing the switching region. When $\alpha^{min} = \alpha^{max}$ this is equivalent to setting $Q = 0$. Thus α^{min} provides an additional way to adjust the responsiveness/vulnerability of the swarm. The results for switching from torus to flock are shown in Figure C.9. These results also show that increasing α^{min} decreases the interaction time and increases the switching region over M.

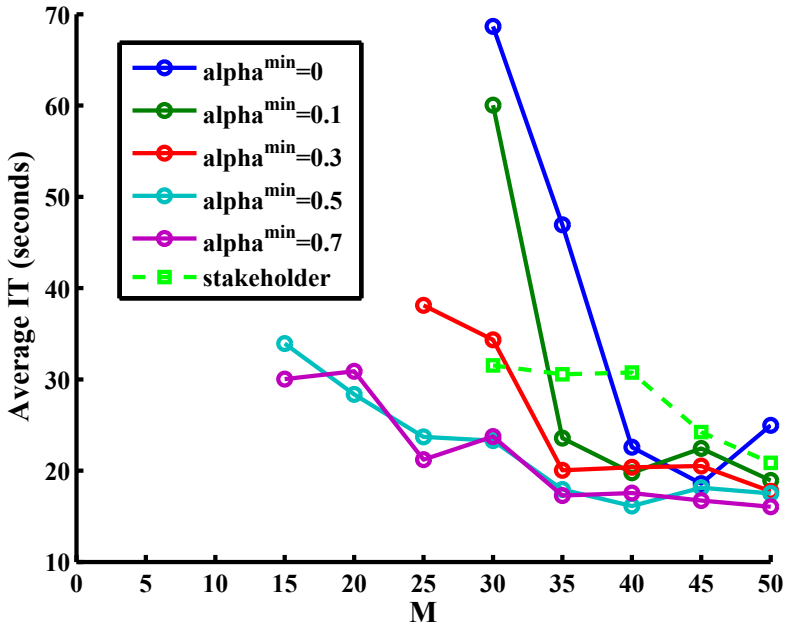
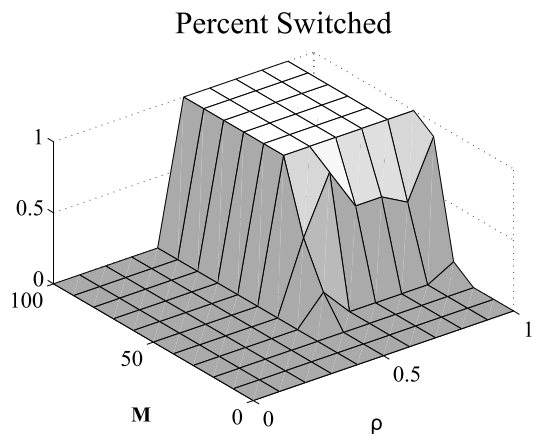


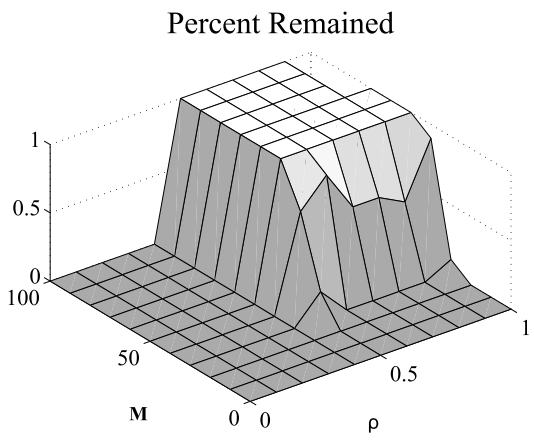
Figure C.9: Average interaction time to switch from torus to flock for stakeholder model, and quorum sensing model for $Q = 2$, $\alpha^{max} = 0.8$, and $0 \leq \alpha^{min} \leq 0.7$.

C.3 Effects of heterogeneous parameters

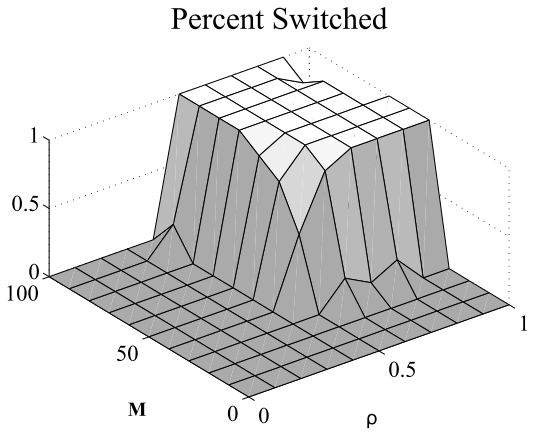
We also investigated the behavior of the quorum sensing model when there is variation in the model parameters. We started by running a set of simulations using the quorum sensing model with homogeneous model parameters. For switching from flock to torus we used $\alpha^{max} = 0.6$, $\alpha^{min} = 0.2$, and $Q = 2$, for all agents. For switching from a torus to a flock we used $\alpha^{max} = 0.8$, $\alpha^{min} = 0.2$, and $Q = 2$, for all agents. The results for the homogeneous parameter sets as M and ρ are varied are shown in figures C.10(a), C.10(b), C.11(a), and C.11(b). To investigate the effects of using heterogeneous parameters we ran the same simulations but with heterogeneous parameters. For switching from flock to torus we used $\alpha_i^{max} \sim \mathcal{N}(0.6, 0.05)$, $\alpha_i^{min} \sim \mathcal{N}(0.2, 0.05)$, and $Q_i \sim DU(0, 4)$, $\forall i$. For switching from torus to flock we used $\alpha_i^{max} \sim \mathcal{N}(0.8, 0.05)$, $\alpha_i^{min} \sim \mathcal{N}(0.2, 0.05)$, and $Q_i \sim DU(0, 4)$, $\forall i$, where $\mathcal{N}(\mu, \sigma)$ is the normal distribution parameterized by mean μ and standard deviation σ, and $DU(a, b)$ is the discrete uniform distribution over integers in the interval $[a, b]$. The results for the heterogeneous parameter sets are shown in figures C.10(c), C.10(d), C.11(c), and C.11(d).



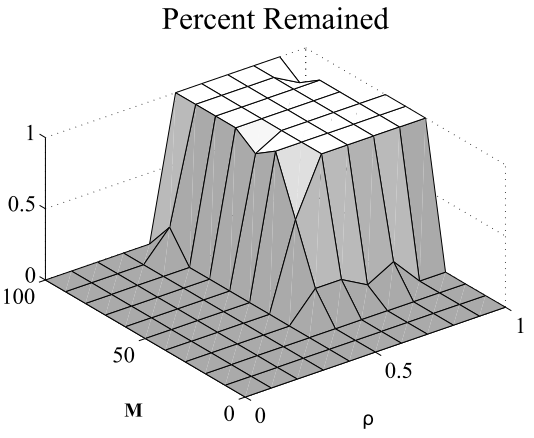
(a) Homogeneous



(b) Homogeneous



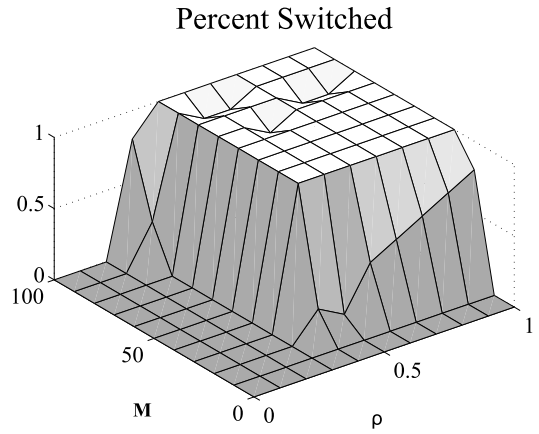
(c) Heterogeneous



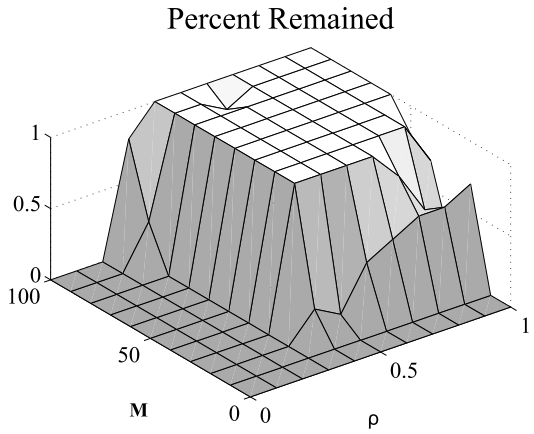
(d) Heterogeneous

Figure C.10: Switch from flock to torus using quorum sensing with homogeneous and heterogeneous parameters.

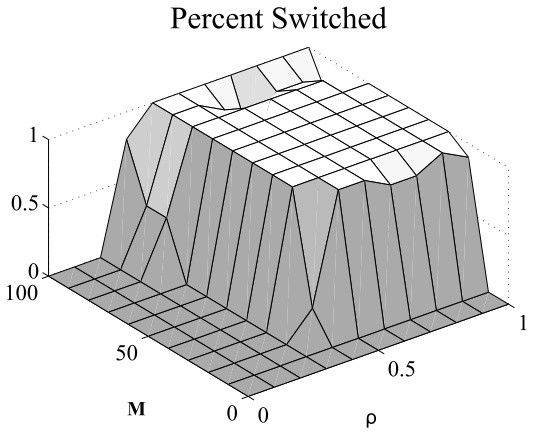
Examining these results we see that the model is quite robust to parameter variation and noise and that adding heterogeneous parameters actually decreases the minimum M needed to allow reliable switching for both switching from flock to torus and switching from torus to flock.



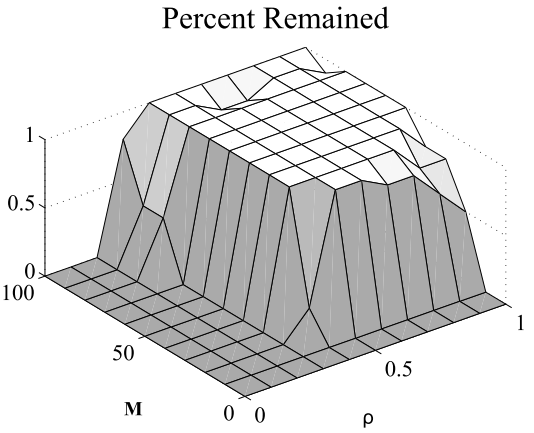
(a) Homogeneous



(b) Homogeneous



(c) Heterogeneous



(d) Heterogeneous

Figure C.11: Switch from torus to flock using quorum sensing with homogeneous and heterogeneous parameters.

Appendix D

Maximum Speed of a Torus

This section investigates the theoretical upper limit on the speed of a torus. Although the torus group type is usually relatively stationary, there is always a small random drift to the torus. Additionally, if human or environmental influences affect the individual agents in the torus, the torus centroid may change more rapidly. If we wish to have the torus move in some direction it is helpful to know how fast the torus can move to determine whether switching to a flock formation is necessary. Additionally, if we are unsure of the collective behavior of the agents in the swarm, but have an estimate of the velocity of the group centroid, then we could compare this estimate with the upper limit on the speed of a torus to gain insight into the group type of the swarm.

We can think of the torus formation as a rotating disc of radius r. If we consider one agent moving along the perimeter of the torus and if we assume the torus is itself moving (i.e. the centroid of the torus has a certain velocity) then we can think of the agent's trajectory as the cycloid shown in Figure D.1.

The parametric equations that govern the motion of a cycloid generated by a circle of radius r and parameterized by θ, the angle through which the rolling circle has rotated, are

$$x = r(\theta - \sin\theta) \tag{D.1}$$

$$y = r(1 - \cos\theta). \tag{D.2}$$

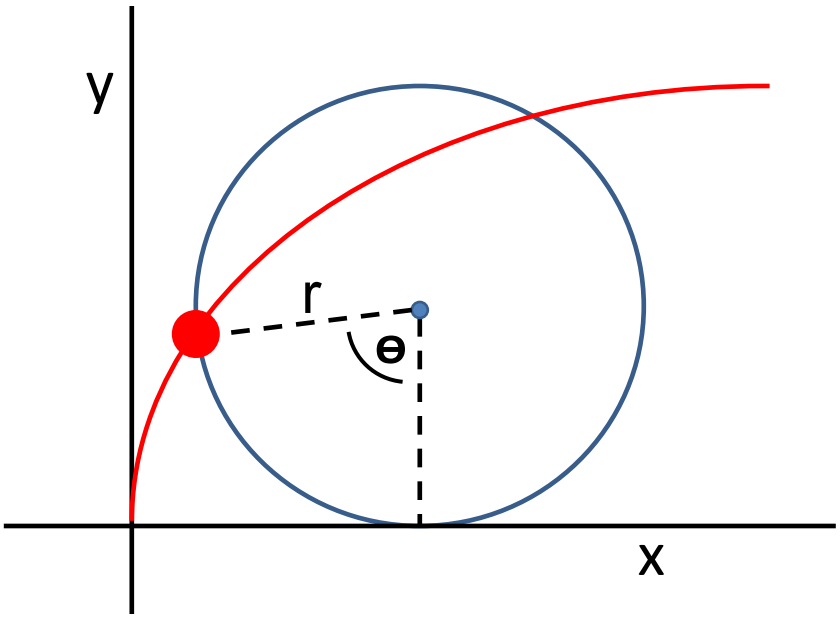


Figure D.1: Idealized trajectory of an agent represented as a cycloid generated by a circle of radius r that has rotated through the angle θ.

To determine the distance that the agent travels we must calculate the length of the parametric curve defined by D.1 and D.2 above. The arc length of a parametric curve, L, parameterized by θ is defined as

$$L = \int_a^b \sqrt{(\frac{dx}{d\theta})^2 + (\frac{dy}{d\theta})^2}\ d\theta. \tag{D.3}$$

So, the length of one arch of the cycloid is

$$\begin{aligned} L &= \int_0^{2\pi} \sqrt{(r(1-\cos\theta))^2 + (r\sin\theta)^2}\ d\theta \\ &= r\int_0^{2\pi} \sqrt{2-2\cos\theta}\ d\theta \\ &= r\int_0^{2\pi} \sqrt{4\sin^2\frac{\theta}{2}}\ d\theta \\ &= 8r. \end{aligned}$$

Assuming that the agent has a speed of s units per second, we want to find the speed of the centroid of the torus, s_{torus}. The time taken for the agent to traverse the arc length is $8r/s$. The center of the disc, or centroid of the torus, has traveled $2\pi r$ units, therefore

$$\max(s_{torus}) = \frac{2\pi r}{8r/s} = \frac{\pi}{4}s. \tag{D.4}$$

Thus, the upper limit on the speed of a torus is approximately three-fourths the speed of an individual agent.

Appendix E

Discrete-Time Approximation Error for Attraction-Only Dynamics

In this chapter we examine the error between our discrete-time approximation and the actual continuous dynamics for a the simplified case of attraction only dynamics. We show that the error between the continuous-time model and the discrete-time approximation can be quantified in terms of the time step, the agent speed, and the gain on an agent's angular velocity. We will also show that the radius of the torus can be controlled using the gain on agent angular momentum.

We consider the behavior of a swarm with continuous attraction-only dynamics

$$\dot{c}_i = s \cdot v_i \tag{E.1}$$

$$\dot{\theta}_i = w_i \tag{E.2}$$

where $c_i = [x_i, y_i]^T \in \mathbb{R}^2$ is the i^{th} agent's position, $v_i = [\cos(\theta_i), \sin(\theta_i)]^T$ is a unit velocity vector for agent i, $\theta_i \in [-\pi, \pi]$ is the angular heading of the agent, s is the constant agent speed, and w_i is the angular velocity.

Because we are restricting our analysis to attraction-only dynamics, the desired heading vector u_i is computed as

$$u_i = u_i^a. \tag{E.3}$$

Therefore, the angular velocity, w_i, is

$$w_i = k\big(\theta_i^d - \theta_i\big) \tag{E.4}$$

where $\theta_i^d = \text{atan2}(u_i^a)$ is the desired orientation of agent i. If we assume a complete topology, then $\theta_i^d = \text{atan2}(r_{ic})$ where $r_{ic} = c_g - c_i$ and c_g is the group centroid. As shown previously, the resulting group behavior is a cyclic group with radius of

$$r = \frac{2s}{k\pi}. \tag{E.5}$$

E.1 Approximation error

The discrete-time dynamics of our system are

$$c_i(t+1) = c_i(t) + s \cdot v_i(t)\Delta t \tag{E.6}$$

$$\theta_i(t+1) = \theta_i(t) + k\alpha_i(t)\Delta t \tag{E.7}$$

where c is the agent position, $\alpha_i(t) = \theta_i^d(t) - \theta_i(t)$, (adjusted so the agent moves in the appropriate direction) and Δt is the discretized time step.

Because this is only an approximation of the continuous dynamics we want to quantify the error between our approximation and the actual continuous model. As the simulation time step Δt approaches 0, the simulation values will match the continuous values described above. For the discrete case, the agent trajectories will be discrete approximations of the circle and can therefore be viewed as regular polygons inscribed inside of a circle. The length of the sides of the polygon l will be the distance traveled at each time step so $l = s\Delta t$. At each time step the agent will compute α, change its direction to be closer to θ^d and then move a distance of l in the direction $\theta(t+1)$.

Given an n-gonal approximation of a circle we can use a little trigonometry to calculate the radius of the circle r. The radius of this circle will also be the radius of the orbit of an agent following the n-gonal discrete-time orbit. Consider the diagram in Figure E.1. Given the number of sides n we have that

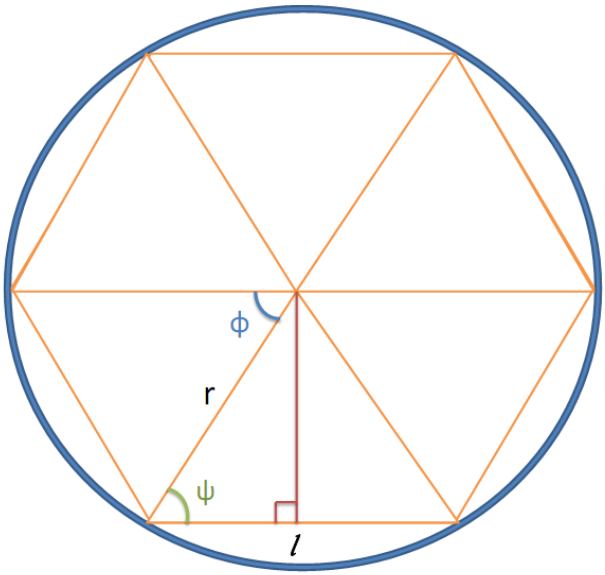


Figure E.1: N-gonal approximation of a circle

$$\phi = \frac{2\pi}{n}. \tag{E.8}$$

Examining the right triangle formed by drawing a line from the center of the polygon perpendicular to one of the edges we have that

$$\frac{l}{2} = r \sin \frac{\phi}{2} = r \sin \frac{\pi}{n}. \tag{E.9}$$

As previously noted, the agent moves $l = s\Delta t$ units per time step. Substituting this for l in Equation E.9 and solving for r we have

$$r = \frac{s\Delta t}{2 \sin(\pi/n)} \tag{E.10}$$

and conversely, solving for n we have

$$n = \frac{\pi}{\sin^{-1}(s\Delta t/2r)}. \tag{E.11}$$

These equations require that we know n in order to compute r and vice versa. We would like an equation that only depends on Δt, α, and s. To achieve this goal we first note that the time taken for an agent to complete an orbit of the centroid is

$$t_{\text{oribit}} = \frac{2\pi}{k\alpha} \tag{E.12}$$

and that the distance traveled by the agent is

$$d = s \cdot \frac{2\pi}{k\alpha}. \tag{E.13}$$

These equations are true regardless of whether the agents move in continuous or discrete time. We also know that the length of a side l equals $s\Delta t$. Therefore the number of sides n is given by dividing d, the perimeter of the n-gon shown in Figure E.1, by l. Thus

$$n = \frac{d}{l} = \frac{2\pi}{k\alpha\Delta t}. \tag{E.14}$$

We now have an equation for n that doesn't depend on r which is good, but we would like to get rid of the α as well. In the continuous case we know that agents in the stable cycle have $\alpha = \pi/2$; however, in the discrete case there will be some error. To calculate the error it helps to consider Figure E.2

An agent's heading is represented in Figure E.2 as a blue vector. The angle $\tilde{\theta}$ represents the difference between the discrete-time approximation and the continuous model. We can solve for the magnitude of $\tilde{\theta}$ by noting that $\phi/2 + \psi = \pi/2$ and also that $\psi + |\tilde{\theta}| = \pi/2$. Therefore

$$|\tilde{\theta}| = \frac{\phi}{2} = \frac{\pi}{n}. \tag{E.15}$$

As agents travel along the n-gonal approximation of the circle, their heading will point in the same direction as the side that they most recently traversed. Because agents in both the

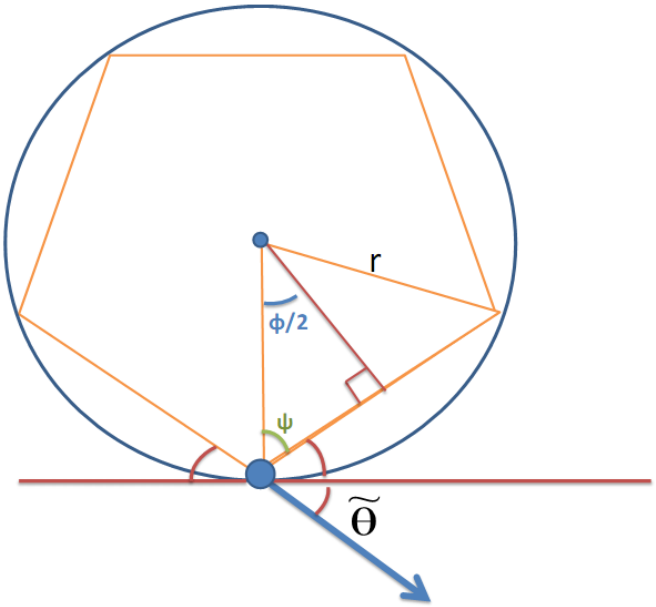


Figure E.2: Error in heading ($\tilde{\theta}$) for discrete-time simulation

discrete-time and continuous-time models are attracted to the centroid, in the continuous case, as stated earlier, $\alpha = \pi/2$ at equilibrium and in the discrete case

$$\alpha = \frac{\pi}{2} + \frac{\pi}{n} \tag{E.16}$$

at equilibrium. Plugging this result into Equation E.14 and solving for n we have the following equation for the number of sides

$$n = \frac{4}{k\Delta t} - 2. \tag{E.17}$$

Plugging this result into Equation E.10 we have the following equation for the radius of the equilibrium cycle of the discrete-time model

$$r = \frac{s\Delta t}{2\sin\left(k\pi\Delta t/(4 - 2k\Delta t)\right)}. \tag{E.18}$$

E.2 Error

We can now calculate the error between the discrete- and continuous-time attraction-only swarm models described in this paper.

$$\tilde{r} = \frac{2s}{k\pi} - \frac{s\Delta t}{2\sin\left(k\pi\Delta t/(4-2k\Delta t)\right)} \tag{E.19}$$

$$\tilde{\alpha} = \frac{\pi}{n} = \frac{k\pi\Delta t}{4-2k\Delta t}. \tag{E.20}$$

E.2.1 Analysis of results

We can verify these results by showing that for the continuous model α is in fact equal to $\pi/2$ when the system reaches a stable cycle. The previous results can be verified quite easily. First we consider our result for θ^d in the discrete simulation

$$\begin{aligned} \theta^d &= \frac{\pi}{2} + \frac{\pi}{n} && \text{(E.21)} \\ &= \frac{\pi}{2} + \frac{\pi}{(4/(k\Delta t - 2))} && \text{(E.22)} \\ &= \frac{\pi}{2 - k\Delta t}. && \text{(E.23)} \end{aligned}$$

If we now take the limit as Δt approaches zero we have

$$\lim_{\Delta t\to 0} \alpha = \lim_{\Delta t\to 0} \frac{\pi}{2-k\Delta t} = \frac{\pi}{2}. \tag{E.24}$$

If we assume a fixed α and s we can also analyze r for the discrete model and let $\Delta t \to 0$.

$$\begin{aligned} \lim_{\Delta t\to 0} r &= \lim_{\Delta t\to 0} \frac{s\Delta t}{2\sin\left(\frac{\pi k\Delta t}{4-2k\Delta t}\right)} && \text{(E.25)} \\ &= \lim_{\Delta t\to 0} \frac{s}{2\cos\left(\frac{k\pi\Delta t}{4-2k\Delta t}\right)} \left(\frac{(4-2k\Delta t)(\pi k) - (k\pi\Delta t)(-2k)}{(4-2k\Delta t)^2}\right)^{-1} && \text{(E.26)} \\ &= \frac{2s}{k\pi} && \text{(E.27)} \end{aligned}$$

which matches our previous analysis.

E.2.2 Controlling the radius of a swarm through the gain

Our final analysis considers the result of changing the gain k on angular velocity.

$$\lim_{k\to 0} r = \lim_{k\to 0} \frac{s\Delta t}{2\sin\left(\frac{k\pi\Delta t}{4-2k\Delta t}\right)} \tag{E.28}$$

$$= \infty. \tag{E.29}$$

If we take the limit to infinity we get

$$\lim_{\alpha\to\infty} r = \lim_{\alpha\to\infty} \frac{s\Delta t}{2\sin\left(\frac{\pi k\Delta t}{4-2k\Delta t}\right)} \tag{E.30}$$

$$= \frac{s\Delta t}{2\sin\left(\frac{\pi}{-2}\right)} \tag{E.31}$$

$$= -\frac{s\Delta t}{2}. \tag{E.32}$$

This value is negative and therefore is not realizable. In fact by examining r we see that there are discontinuities whenever

$$\sin\left(\frac{k\pi\Delta t}{4-2k\Delta t}\right) = 0 \qquad \text{or} \qquad 4-2k\Delta t = 0. \tag{E.33}$$

Solving the latter equation we find that if $k = 2/\Delta t$ there will be a discontinuity. The former will be zero when $\frac{k\pi\Delta t}{4-2k\Delta t} = n\pi$ for $n = 0, \pm 2, \pm 4, \cdots$. Therefore there will be discontinuities for

$$k = \frac{4n}{(1+2n)\Delta t} \quad \text{for} \quad n = 0, \pm 2, \pm 4, \cdots . \tag{E.34}$$

E.3 Simulation results and validation

In order to validate this analysis, we ran a set of simulations using different values for Δt with $k = 1$ and $s = 1$. Each simulation was run for 200 seconds and 50 agents were used.

	Simulation			Theoretical		
Δt	r	θ^d	sides	r	θ^d	sides
.5	0.5	2.0944	6.0	0.5	2.0944	6
.4	0.5226	1.9635	8.0	0.5226	1.9635	8
.3	0.5481	1.8480	11.333	0.5481	1.8480	11.333
.2	0.5759	1.7453	18.0	0.5759	1.7453	18
.1	0.6055	1.6535	38.0	0.6055	1.6535	38
.01	0.6334	1.5787	397.99	0.6334	1.5787	398
.001	0.6363	1.5716	3998.0	0.6363	1.5716	3998
ϵ				0.6366	1.5708	∞

Table E.1: Comparison of simulated results with theoretical results. The last row shows the values for the continuous case.

The results are shown in Table E.1. These results show that the simulation results match the theoretical results.

The results were calculated at the end of each simulation. The radius was calculated by finding the average agent distance from the group centroid. The desired direction change α was calculated by finding the average of $|k(\theta^d - \theta)|$ for all agents. The number of sides of the n-gonal approximation was calculated using Equation (E.11). The theoretical results for r, α, and the number of sides were calculated using equations (E.23), (E.18), and (E.17), respectively.

E.4 Dependence on initial conditions

The above results all depend on the centroid converging to a constant point. When experimenting with five agents, we noticed that quite often the centroid would not converge, but would itself travel in a polygonal trajectory. This appears to be related to the initial conditions of the agents—if the agents are unbalanced, then the centroid will not converge to a point, but will itself converge to a sort of limit cycle (think of a rolling oval and how its centroid moves in a circle).

We tracked the centroid for one of the simulations that didn't converge to a point and the trajectories are shown in Figure E.3. This simulation used $\Delta t = .5$ which, as shown in

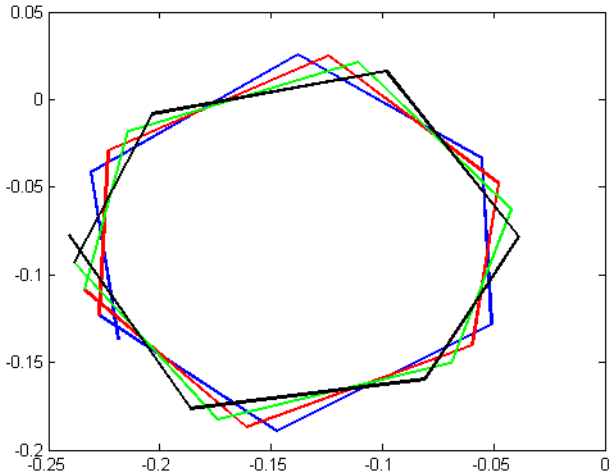


Figure E.3: Trajectories of a centroid for 5 agents with $\Delta t = .5$. Colors are used to make the hexagonal shape more apparent.

Table E.1, results in hexagonal agent trajectories. Figure E.3 suggests that not only do the agents move in hexagonal trajectories, but if the centroid does not converge, then it also moves in a hexagonal trajectory.

E.5 Discussion

We have demonstrated that the radius of a swarm can be calculated for both the continuous and discrete cases. The error between the discrete case and the continuous case can be calculated using the results shown in this paper. Future work should investigate calculating the discrete-time approximation error for the full swarm model, and also investigate whether the end behavior of the centroid can be determined if the initial conditions are known.

References

[1] D. Sumpter, "The principles of collective animal behaviour," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 361, no. 1465, pp. 5–22, 2006.

[2] M. Steinberg, "Biologically-inspired approaches for self-organization, adaptation, and collaboration of heterogeneous autonomous systems," in *Proceedings of SPIE Volume*, vol. 8062, 2011.

[3] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

[4] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[5] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[6] I. Couzin, J. Krause, N. Franks, and S. Levin, "Effective leadership and decision-making in animal groups on the move," *Nature*, vol. 433, no. 7025, pp. 513–516, 2005.

[7] L. Conradt, J. Krause, I. Couzin, and T. Roper, "Leading according to need in self-organizing groups," *The American Naturalist*, vol. 173, no. 3, pp. 304–312, 2009.

[8] I. Couzin, J. Krause, R. James, G. Ruxton, and N. Franks, "Collective memory and spatial sorting in animal groups," *Journal of Theoretical Biology*, vol. 218, no. 1, pp. 1–11, 2002.

[9] S. Kerman, D. Brown, and M. Goodrich, "Supporting human interaction with robust robot swarms," in *Proceedings of the International Symposium on Resilient Control Systems*, Aug 2012.

[10] W. Spears, D. Spears, R. Heil, W. Kerr, and S. Hettiarachchi, "An overview of physicomimetics," *Swarm Robotics*, pp. 84–97, 2005.

[11] C. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *ACM SIGGRAPH Computer Graphics*, vol. 21, pp. 25–34, ACM, 1987.

[12] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, 1995.

[13] J. Marshall, M. Broucke, and B. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Transactions on Automatic Control*, vol. 49, no. 11, pp. 1963–1974, 2004.

[14] H. Levine, W. Rappel, and I. Cohen, "Self-organization in systems of self-propelled particles," *Physical Review E*, vol. 63, no. 1, p. 017101, 2000.

[15] D. Strömbom, "Collective motion from local attraction," *Journal of theoretical biology*, vol. 283, no. 1, pp. 145–151, 2011.

[16] L. Romero, E. Forgoston, and I. Schwartz, "Noise, bifurcations, and modeling of interacting particle systems," in *International Conference on Intelligent Robots and Systems*, pp. 3905–3910, IEEE, 2011.

[17] H. Su, X. Wang, and Z. Lin, "Flocking of multi-agents with a virtual leader," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 293–307, 2009.

[18] V. Gervasi and G. Prencipe, "Coordination without communication: The case of the flocking problem," *Discrete Applied Mathematics*, vol. 144, no. 3, pp. 324–344, 2004.

[19] H. LeBlanc and X. Koutsoukos, "Low complexity resilient consensus in networked multi-agent systems with adversaries," in *Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control*, pp. 5–14, ACM, 2012.

[20] M. Wirz, D. Roggen, and G. Troster, "Decentralized detection of group formations from wearable acceleration sensors," in *International Conference on Computational Science and Engineering*, vol. 4, pp. 952–959, IEEE, 2009.

[21] A. Sadilek and H. Kautz, "Location-based reasoning about complex multi-agent behavior," *Journal of Artificial Intelligence Research*, vol. 43, pp. 87–133, 2012.

[22] P. Laube, M. Duckham, and T. Wolle, "Decentralized movement pattern detection amongst mobile geosensor nodes," *Geographic Information Science*, pp. 199–216, 2008.

[23] J. Gudmundsson, M. van Kreveld, and B. Speckmann, "Efficient detection of motion patterns in spatio-temporal data sets," in *Proceedings of the 12th annual ACM international workshop on Geographic information systems*, pp. 250–257, ACM, 2004.

[24] J. Gudmundsson, M. van Kreveld, and B. Speckmann, "Efficient detection of patterns in 2d trajectories of moving points," *Geoinformatica*, vol. 11, no. 2, pp. 195–215, 2007.

[25] N. Eagle and A. Pentland, "Reality mining: sensing complex social systems," *Personal and Ubiquitous Computing*, vol. 10, no. 4, pp. 255–268, 2006.

[26] N. Eagle and A. Pentland, "Eigenbehaviors: Identifying structure in routine," *Behavioral Ecology and Sociobiology*, vol. 63, no. 7, pp. 1057–1066, 2009.

[27] N. Eagle, A. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," *Proceedings of the National Academy of Sciences*, vol. 106, no. 36, pp. 15274–15278, 2009.

[28] L. Tang, X. Wang, and H. Liu, "Scalable learning of collective behavior," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 6, pp. 1080–1091, 2012.

[29] A. Campbell, C. Riggs, and A. Wu, "On the impact of variation on self-organizing systems," in *Fifth IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, pp. 119–128, IEEE, 2011.

[30] A. Campbell and A. Wu, "Multi-agent role allocation: issues, approaches, and multiple perspectives," *Autonomous agents and multi-agent systems*, vol. 22, no. 2, pp. 317–355, 2011.

[31] M. Hsieh, Á. Halász, S. Berman, and V. Kumar, "Biologically inspired redistribution of a swarm of robots among multiple sites," *Swarm Intelligence*, vol. 2, no. 2, pp. 121–141, 2008.

[32] R. Beard and T. McLain, *Small unmanned aircraft: Theory and practice*. Princeton University Press, 2012.

[33] J. P. Hespanha, *Linear systems theory*. Princeton University Press, 2009.

[34] W. Ren and R. Beard, *Distributed consensus in multi-vehicle cooperative control: theory and applications*. Springer, 2007.

[35] D. Olsen and M. Goodrich, "Metrics for evaluating human-robot interactions," in *Proceedings of PERMIS*, vol. 2003, 2003.

[36] S. Kerman, "Metrics for human interaction with bio-inspired swarms," Master's thesis, Brigham Young University, Provo, UT, 2013.

[37] P. Walker, S. Nunnally, M. Lewis, A. Kolling, N. Chakraborty, and K. Sycara, "Investigating neglect benevolence and communication latency during human-swarm interaction," in *AAAI Fall Symposium Workshop on Human Control of Bio-Inspired Swarms*, 2012.

[38] D. Cvetkovic, P. Rowlinson, and S. Simic, *An introduction to the theory of graph spectra*. Cambridge University Press, 2010.

[39] M. Raghib, S. Levin, and I. Kevrekidis, "Multiscale analysis of collective motion and decision-making in swarms: An advection–diffusion equation with memory approach," *Journal of Theoretical Biology*, vol. 264, no. 3, pp. 893–913, 2010.

[40] M. Hayes, *Statistical digital signal processing and modeling*. John Wiley & Sons, 2009.

[41] M. A. Goodrich, B. Pendleton, S. Kerman, P. Sujit, and J. Pinto, "Enabling human interaction with bio-inspired robot teams: Topologies, leaders, predators, and stakeholders," Technical Report, BYU-HCMI Lab, 2011.

[42] M. A. Goodrich, S. Kerman, and S. Y. Jung, "On leadership and influence in human-swarm interaction," in *AAAI Fall Symposium Workshop on Human Control of Bio-Inspired Swarms*, 2012.

[43] L. Conradt and T. Roper, "Consensus decision making in animals," *Trends in Ecology & Evolution*, vol. 20, no. 8, pp. 449–456, 2005.

[44] I. Couzin, "Collective cognition in animal groups," *Trends in cognitive sciences*, vol. 13, no. 1, pp. 36–43, 2009.