Evaluating Adaptive and Non-adaptive Strategies for Selecting and Orienting Influencer Agents for Effective Flock Control

Adam Dees
University of Tulsa
apd615@utulsa.edu

James Hale
University of Tulsa
jah6484@utulsa.edu

Sandip Sen
University of Tulsa
sandip@utulsa.edu

ABSTRACT

Flocks navigate for large distances, moving in a coherent path through space, under mutual influence of flock members. Such influences may include repulsion, orientation, and attraction. Certain applications give rise to the need to control the movements of flocks, e.g., circumventing critical zones. Researchers have investigated the problem of seeding flocks with a percentage of externally controlled agents to achieve effective flock control. Recent studies of flock control include orthogonal directions of (a) selecting influencing or leader agents and (b) orienting the leader agents. We build on these studies and evaluate combinations of selecting and orienting choices for fast convergence of the flock to follow desired travel directions with both adaptive and non-adaptive selection and orientation algorithms. We evaluate the effectiveness of combined flock control strategies under different physical world models.

1 INTRODUCTION

Multiagent system researchers are interested in designing and analyzing ad hoc and emergent coordination among agents. Of particular interest to us is the topic of synthesizing coordinated behavior in groups without any explicit communication or prior agreements such as abiding by coordination protocols. Coordination in such scenarios emerges from interaction or behavioral rules followed by agents in a group. Individual agents in such groups, often referred to as flocks, swarms, or herds, use innate behaviors to respond to sensory stimuli from neighboring agents. Sensory stimuli can include positions, orientations, velocities, etc. of neighboring agents. Behavioral influences from other agents include repulsion, attraction, orientation, etc. that shape the movement of each agent in addition to any other existing environmental influences.

In this paper, we use the term flocks and flocking to refer to a range of coordination scenarios and behaviors that include situations which have been traditionally referred to as flocking, swarm control, herding behaviors, etc. in the literature [25]. Researchers in diverse disciplines, e.g., physicists [31], biologists, and computer graphicists [25], among others, have investigated such behaviors. More often than not, researchers in natural sciences are interested in observing, analyzing and explaining emergent properties of flocking behaviors in the animal kingdom. A wide range of life forms exhibit flocking behaviors, ranging from insects such as swarms of bees, shoals of fish [12, 22], flocks of birds [20, 24, 27, 34], herds of existing (elephants, giraffes, deers, primates [28], etc.) and even extinct (mammoths [14]) animal species, etc. Incentives for flocking are equally diverse and includes improved foraging [24, 27], defense against predators [2, 13], navigational efficiency [12, 22], protection against environmental hazards [4], survival of abandoned chicks in 'creches' [20, 34], etc.

Agent-based simulation is a valuable tool for natural scientists to complement empirical research that relies on data gathered from observations. For example, one can envision the use of representative simulations of animal flocks to study the effects of intervention mechanisms, e.g., for herding an endangered population from regions of depleted resources or targets of illegal poaching to safer, more abundant and sustainable location. High-fidelity simulations of flocks, effectively informed by meticulous empirical research, can be a valuable tool for flock management by conservationists, environmental workers, wildlife biologists, herders, etc. In a reverse fashion, computer scientists have developed effective optimization algorithms inspired by empirically observed flocking and swarming behaviors [5, 17, 18, 26].

We, like other multiagent system researchers [6–11, 15, 29, 30, 32], however, are interested in studying flocks of artificial agents, either robotic or virtual. The generic task that we investigate is the control of a large group of artificial agents whose behavioral characteristics are well-documented, i.e., we know how these agents respond to sensory stimuli. The research goal is to strategically place a few carefully designed influencer or leader agents in the flock to control flock movement to follow the desired trajectory. Motivating applications for the use of influencer agents for flock control include guiding a flock of birds around critical areas, such as airports and wind farms, where they would otherwise have the potential to do substantial damage to humans and themselves, without veering far from the course. If relatively few leaders can be shown to effectively and robustly maneuver large flocks, agent designers can rapidly deploy targeted solutions in a short time, by programming only a few leader agents, for new goal trajectories or orientation. The large majority of the agents in the flock, the non-leader or follower agents, need not be reprogrammed for a new "mission". The ability to effectively control a large group of follower agents, with relatively few leader agents possessing mission-specific control behaviors, would provide significant cost savings and timely responses to new mission goals in comparison to reprogramming and deploying every agent in the flock.

Prior work on agent-based flock control have investigated various key aspects including the following:

- **Leader Selection:** Various strategies have been investigated to either select leaders from the available agents [7, 30] or place new leaders in locations that will maximize their ability to control the flock [6, 9].
- **Leader Orientation:** Various behaviors have been investigated to determine the direction and timing of orientation and velocities to be chosen by the leaders to most effectively control the flock [6, 9].
- **Leader Roles:** Some research has investigated a democratic process of electing leaders periodically and hence leadership

can change over time [32] and leaders can be mediators, controlled by human users, who perform diverse roles in the flock [16].

Other Leader Behaviors: In certain situations, such as under sparse and distributed agent scenarios, leaders may first strategically position themselves in the group, which might include waiting in a holding pattern, before triggering their influencing orientation behaviors [6].

Human Control: Human controllers can exert either indirect control through changing environmental features [19, 33] or direct control by changing agent parameters [3, 25], providing intermittent or continuous inputs [1, 23].

Influence Models: An influence model determines how agents are influenced by other agents in their neighborhood. Influence models vary in terms of the effect of distance of the neighbor and a subset of attractive, repulsive and orientation effects exerted on an agent by its neighbors [16, 30] (see Figure 1). We note that the influencer agents only exert influence on other agents but are not influenced by others. However, follower agents cannot distinguish between influencer agents and other follower agents and react equally to influences from both of these agent types. It is typically assumed though, that the influencer agents are aware of the location and identity of other influencer agents. This makes the flock control problem more challenging compared to the scenario where the influencers have distinctive positions of hierarchy in the group and hence have a greater ability to influence the followers.

Most of the existing literature on flock control, with few exceptions [9], primarily focus on investigating options for only one aspect of flock control while fixing the other attributes of the agent or the domain. We investigate combinations of the most promising influence selection and orientation strategies and identify some novel approaches that perform well in various environments. We also experiment with multiple influence models, environments or physical realities governing how agents influence one another. The goal is to come away with recommendations for influencer selection and orientation behaviors for some of the commonly investigated influence models.

In particular, we are interested in studying the relative advantages of adaptive versus non-adaptive algorithms. In the current context, *non-adaptive strategy* choices by influencer agents ignore the positions and the likely decisions of other influencer agents, while *adaptive* strategy choices do take those factors into consideration. While the former can be computationally cheaper and requires less sensory and processing capabilities, the latter might be more robust and effective. Our results indicate that different influencer selection and orientation behaviors are best suited for different environments (influence models).

The rest of the paper is organized as follows: Section 2 describes the influence models, adaptive and non-adaptive influencer selection and orientation selection strategies; Section 3 presents the experimental framework and results from the experiments; Section 5 summarizes our findings and identifies future research directions.

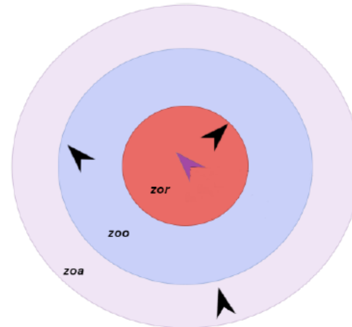

Figure 1: The zonal influence model showing zones of repulsion (zor), orientation (zoo), and attraction (zoa). Slightly modified figure from Tiwari *et al.* [30].

2 MODEL

A flock is composed of a set of influencer agents $\mathcal{I}$ and follower agents $\mathcal{F}$. Each agent a has a two-dimensional position vector $\mathbf{p}_a$ and a two-dimensional velocity vector $\hat{\mathbf{v}}_a$, the latter vector always has a magnitude equal to 1 to maintain constant and equal speed for all agents. Every agent has three concentric circular zones, defined in order of increasing size, one of which is a circle called the "zone of repulsion", $\mathcal{Z}_{R,a}$, and the other two are annuli with inner radii set such that they do not overlap with the zones smaller than them called the "zone of orientation", $\mathcal{Z}_{O,a}$, and the "zone of attraction", $\mathcal{Z}_{A,a}$ (see Figure 1). These agents exist in a looping 2-dimensional space $[0,1]x[0,1]$ in that $\forall i, \mathbf{p}_{a,i} \in [0,1]$. Looping is achieved by checking each component of the position vector and adding 1 so long as the component is less than 0 and subtracting 1 so long as the component is greater than 1. Our distance finding algorithm $d(a,b)$ takes this looping into account by taking the Euclidean norm $||\mathbf{p}_a - \mathbf{p}(a,b)||$, where $\mathbf{p}(a,b)$ is the looping position function, defined as:

$$\mathbf{p}(a,b) = \underset{\mathbf{p}_b+(n,m)}{\arg\min} \; ||\mathbf{p}_a - \mathbf{p}_b + (n,m)||, n \in \{-1,0,1\}, m \in \{-1,0,1\}$$

We initialize flocks by setting agent positions $\mathbf{p}$ uniformly within a circle of radius 0.2 centered around the position (0.5, 0.5). We consider two initial flocking configurations: aligned and unaligned corresponding to a coordinated or uncoordinated flock respectively. Velocity vectors $\hat{\mathbf{v}}_a$ are assigned in the unaligned case by randomly and uniformly selecting a point on the unit circle. In the aligned case, a similar random point on the unit circle is selected and as the midpoint of an arc of length .7 that is uniformly sampled to select initial velocities for all agents. Influencer agents $\mathcal{I}$ are then selected according to one of several algorithms described below, leaving the remainder as follower agents $\mathcal{F}$. A target velocity $\theta_f \sim \mathcal{U}(0, 2\pi)$ for the flock is randomly selected.

The simulation progresses in discrete time-steps which are divided into a phase of velocity update and a phase of position update. During the velocity update phase, each agent chooses its desired next orientation. All agents ultimately update their velocities primarily due to the conditions of their zones of repulsion. Velocity update rules for follower agents vary depending on the influence model. Influencer agents have velocity update rules which vary depending on the particular influencer orientation strategy used. However, only in one of the four cases of the influencer agent velocity update function (given below) is the specific orientation strategy algorithm used.

$$\hat{\mathbf{v}}_a(t+\Delta t) = \begin{cases} -\frac{\sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a}{|| \sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a ||} & \mathcal{Z}_{R,a} \neq \emptyset \\ OrientationStrategy & \mathcal{Z}_{R,a} = \emptyset \wedge \mathcal{Z}_{O,a} \neq \emptyset \\ \frac{\sum_{b \in \mathcal{Z}_{A,a}} \mathbf{p}(a,b) - \mathbf{p}_a}{|| \sum_{b \in \mathcal{Z}_{A,a}} \mathbf{p}(a,b) - \mathbf{p}_a ||} & \mathcal{Z}_{R,a} = \mathcal{Z}_{O,a} = \emptyset \wedge \mathcal{Z}_{A,a} \neq \emptyset \\ \hat{\mathbf{v}}_a(t) & Otherwise \end{cases}$$

All velocities changes (both influencer and follower) are bounded by a maximum turn of $\omega = 0.2°$ per time-step toward their desired velocity. Once velocities are processed, we update the positions of agents accordingly:

$$\mathbf{p}_a(t + \Delta t) = \mathbf{p}_a(t) + \Delta t * \hat{\mathbf{v}}_a(t + \Delta t)$$

These new positions are checked to see if they are out of bounds, and if so they are looped to the opposite side of the plane following the procedure described earlier.

At this point the time-step is complete and we test for convergence of the flock. We define convergence similar to that used by Tiwari et al. [30]. The sum of follower agent Cartesian velocities is converted into a polar coordinate form, then taking the radian component to be θ: representing the average direction of the follower agents. If $\theta \in \theta_f \pm 0.1 radians$ for three time-steps in a row, we consider the flock to have converged. If the flock has not converged yet, another time-step is executed.

2.1 Influence Models

Now we define three different environments corresponding to different ways in which agents exert influences one another.

Distinct zonal: Follower agents in this model are either repelling from, orienting with, or attracting to other agents. These are discrete cases where repelling has ultimate priority, then orientation, and lastly attraction. The follower velocity update function is defined as such:

$$\hat{\mathbf{v}}_a(t+\Delta t) = \begin{cases} -\frac{\sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a}{|| \sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a ||} & \mathcal{Z}_{R,a} \neq \emptyset \\ \frac{\sum_{b \in \mathcal{Z}_{O,a}} \hat{\mathbf{v}}_b}{|| \sum_{b \in \mathcal{Z}_{O,a}} \hat{\mathbf{v}}_b ||} & \mathcal{Z}_{R,a} = \emptyset \wedge \mathcal{Z}_{O,a} \neq \emptyset \\ \frac{\sum_{b \in \mathcal{Z}_{A,a}} \mathbf{p}(a,b) - \mathbf{p}_a}{|| \sum_{b \in \mathcal{Z}_{A,a}} \mathbf{p}(a,b) - \mathbf{p}_a ||} & \mathcal{Z}_{R,a} = \mathcal{Z}_{O,a} = \emptyset \wedge \mathcal{Z}_{A,a} \neq \emptyset \\ \hat{\mathbf{v}}_a(t) & Otherwise \end{cases}$$

Inverse zonal: All zones are active but the influence of other agents is inversely proportional with distance from the agent by an inverse cubed law. With the addition of the inverse distance weighting, this model is equivalent to distinct zonal with all three zone equations summed and normalized. This is a novel model.

Couzin et al. zonal: The orientation and attraction zones work in tandem when the follower agent is not forced to repulse itself from other agents. It is designed such that each agent is equally weighted, as opposed to equally weighting each zone. The follower velocity update function is defined as such:

$$\hat{\mathbf{v}}_a(t+\Delta t) = \begin{cases} -\frac{\sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a}{|| \sum_{b \in \mathcal{Z}_{R,a}} \mathbf{p}(a,b) - \mathbf{p}_a ||} & \mathcal{Z}_{R,a} \neq \emptyset \\ \hat{\mathbf{v}}_a(t) & \mathcal{Z}_{R,a} = \mathcal{Z}_{O,a} = \mathcal{Z}_{A,a} = \emptyset \\ \frac{\sum_{b \in \mathcal{Z}_{O,a}} \hat{\mathbf{v}}_b + \sum_{b \in \mathcal{Z}_{A,a}} \frac{\mathbf{p}(a,b) - \mathbf{p}_a}{||\mathbf{p}(a,b) - \mathbf{p}_a||}}{|| \sum_{b \in \mathcal{Z}_{O,a}} \hat{\mathbf{v}}_b + \sum_{b \in \mathcal{Z}_{A,a}} \frac{\mathbf{p}(a,b) - \mathbf{p}_a}{||\mathbf{p}(a,b) - \mathbf{p}_a||} ||} & Otherwise \end{cases}$$

2.2 Influencer Selection Strategies

Center: Center placement attempts to make the most connected and central agents influencer agents by first discarding the convex hull of the agents from consideration. Afterward it selects the most central $|\mathcal{I}|$ agents according to $|\mathcal{Z}_{O,a} \cup \mathcal{Z}_{R,a}|$ to be influencer agents. This selection algorithm is considered non-adaptive as it assigns leadership to agents near the center, without considering the proximity of other leaders.

Periphery: The objective of this placement is to border the flock of agents with the influencer agents. The method achieves this by repeated taking the convex hull of the agent positions and setting those positions to be influencer agents so long as we still have influencer agents to place. If the convex hull size happens to be greater than the number of influencer agents still to place, we randomly select an equal number of positions from this hull to be influencer agents. This algorithm is considered non-adaptive as it randomly assigns leadership to agents on the convex-hull.

k-Means: The k-Means clustering algorithm [21] is used to select k leaders located near the center of different clusters of agents on the plane. This selection algorithm is adaptive as leader selections affect each other.

Attraction-Repulsion (AB): The motivation for this approach is to select leaders that are close to clusters of agents while still being somewhat evenly dispersed in the population so that almost all follower agents are being influenced. To achieve this, k target positions for influencers are identified that balance attractive forces from all follower agents, weighted by an attraction parameter, A, and repulsive forces from all other target positions, weighted by a repulsion parameter, R. Subsequently the nearest agent to each of the identified points is chosen as the influencer agents. This algorithm is adaptive as it considers the locations of other selected leaders when assigning leadership roles. The algorithm, developed for this paper, is presented in detail within Algorithm 1.

2.3 Influencer Orientation Strategies

Face Target Direction (FT): The simplest of all the orientation algorithms; all influencers orient towards the target direction. This is the only non-adaptive orientation algorithm in our experiments, as the agent's actions are not influenced at all by the behavior of other members of the flock.

One-Step Lookahead (OSL): Each influencer agent performs a one-step lookahead to check, for each of its possible orientations, the resultant orientation of the followers around it

```
Input :
    α ← .3: Attraction value
    β ← .7: Repulsion value
    I: Set of influencer agents (initially randomly assigned)
    F: Set of following agents
I′ ← I
for c ∈ I′ do
    vectorSumInfluencers ← (0, 0)
    vectorSumFollowers ← (0, 0)
    for a ∈ F ∪ I do
        if a ∈ F then
            vectorSumInfluencers ←
              vectorSumInfluencers − p_a + p_c
        else
            vectorSumFollowers ←
              vectorSumFollowers + p_a − p_c
    point ← p_c + (vectorSumInfluencers ∗ β/|I|) +
      (vectorSumFollowers ∗ α/|F|)
    newLeader ← arg min_{a∈F} d(a, point)
    I ← I − {c} ∪ {newLeader}
    F ← F ∪ {c} − {newLeader}
```

Algorithm 1: The Attraction-Repulsion (AR) influencer selection algorithm.

after one step. The influencer then chooses the orientation that results in the closest alignment of the followers one step into the future. See Algorithm 2 for details. This algorithm is adaptive as an agent's decisions are determined by the movements of surrounding agents.

Augmented OSL (AOSL): A slightly revised form of one-step lookahead which will not consider influencer agents in its own orientation zone; precisely, this considers agents $b \in \mathcal{Z}_{O,a} \cap \mathcal{F}$ instead. This is because OSL calculates the orientations of those in the $\mathcal{Z}_{O,b}$ as if $\hat{\mathbf{v}}_b$ will tend toward *choice* should a adopt it, which does not necessarily follow for another influencer agent b. Like OSL, this algorithm is adaptive, for the same reasoning as OSL.

Two-step Lookahead (TSL): A sophisticated version of one-step lookahead which considers two round in the predicted future to make one of the n^2 choice combinations that minimize the θ_f error of those around it in the next two steps. Again, this algorithm is adaptive.

Coordinated: A OSL rooted implementation that creates pairings between two influencer agents. No influencer agent is allowed to belong to more than one pairing, although an influencer agent could be without one. Pairings were globally maximized based on the amount of intersection between two influencers' zones of orientation. The two members of the pair would consider every combination of choices it and its partner could make, and minimize the θ_f error of agents in both its and its partner's zone of orientation. This algorithm depends on other agents' actions, so it is labeled adaptive.

```
Input :
    Z_{O,a}: Zone of orientation of a
    θ_f: Target theta
    n: Number of angles to consider
    F: Set of follower agents
velocityChoices ← {2π i/n | i ∈ {0, 1, ··· , n}}
bestError ← ∞
bestChoice ← ∅
for choice ∈ velocityChoices do
    orients ← ∅
    for b ∈ Z_{O,a} do
        orient ← (0, 0)
        for c ∈ Z_{O,b} do
            if c ∈ F then
                orient ← orient + v̂_c
            else
                orient ← orient + choice
        orients ← orients ∪ {orient}
    errorSum ← 0
    for orient ∈ orients do
        errorSum ←
          errorSum + radianError(toRadians(orient), θ_f)
    meanError ← errorSum/|orients|
    if meanError < bestError then
        bestError ← meanError
        bestChoice ← choice
return bestChoice
```

Algorithm 2: The One-Step Lookahead influencer orientation algorithm.

Adaptive Orientation	Non-Adaptive Orientation
One-Step Lookahead Augmented OSL Two-step Lookahead Coordinated	Face Target Direction

Table 1: Classification of Orientation Algorithms

Adaptive Selection	Non-Adaptive Selection
k-Means Attraction-Repulsion	Center Periphery

Table 2: Classification of Selection Algorithms

2.4 Adaptive vs. Non-Adaptive Approaches

We were interested in the effectiveness of adaptive selection and orientation algorithms when compared to non-adaptive models for flock control. In our experiments, an adaptive behavior is one which alters its behavior based on the state of other agents; while a non-adaptive behavioral model dictates an agent ignore the actions of other agents in its vicinity. We have grouped the orientation and selection algorithms in this paper into groups of adaptive and non-adaptive in Tables 1 and 2 respectively and investigate the relative performance between them in our discussion.

Symbol	Parameter	Default Value
$\mathcal{F}$	Follower agents	$\lvert\mathcal{F}\rvert = 180$
$\mathcal{I}$	Influencer agents	$\lvert\mathcal{I}\rvert = 20$
$\mathcal{A}$	Agents (the flock)	$\mathcal{F} \cup \mathcal{I}$
a	Agent a	
$\mathbf{p}_a$	a's position vector	$\forall i, \mathbf{p}_{a,i} \in [0, 1]$
$\hat{\mathbf{v}}_a$	a's velocity vector	$\lVert\hat{\mathbf{v}}_a\rVert = 1$
$\mathbf{p}(a, b)$	Looping position of b for a	
$d(a, b)$	Distance between a and b	
$\mathcal{Z}_{R,a}$	a's Zone of Repulsion	$\{b \in \mathcal{A} \mid d(a, b) \in [0, 0.01]\} - \{a\}$
$\mathcal{Z}_{O,a}$	a's Zone of Orientation	$\{b \in \mathcal{A} \mid d(a, b) \in (0.01, 0.1]\}$
$\mathcal{Z}_{A,a}$	a's Zone of Attraction	$\{b \in \mathcal{A} \mid d(a, b) \in (0.1, 0.13]\}$
θ_f	Target angle	$\theta_f \sim \mathcal{U}(0, 2\pi)$
Δt	Time-step	.001
ω	Max Turn per time-step	2°
n	Influencer angle options	16

Table 3: Symbols and Corresponding Parameters

3 EXPERIMENTS

All experiments used parameters as listed in Table 3.

Coordinated and TSL orientation strategies did not emerge as practical algorithms for experimentation. Both included substantial computational costs, with coordinated being unable to run feasibly due to the nature of maximizing the connections in a pairing. Genter and Stone perform a brute-force algorithm to find the best possible set of pairs [7]. This is sufficient in low-densities of influencer agents but becomes impossible in higher-density conditions such as those found our default parameters. Even optimization techniques such as branch and bound are unable to recover the strategies. TSL was not included due to its substantial increase in computational cost and lack of significant convergence time reduction compared to other strategies. TSL was, however, able to significantly outperform all but OSL and AOSL (which it outperformed slightly) in less restrictive conditions of infinite turning speed ($\omega = \pi$) and omission of the zones of repulsion and attraction. These conditions are those found in Genter and Stone where TSL performed very well [7].

Every combination of influence model, influencer selection, and influencer orientation strategy was tested for both initially aligned and unaligned conditions. Data presented in Tables 4, 5, and 6 present flock convergence time means and standard deviations averaged over 500 simulations for each of these cases, along with the average performance of particular strategies across combinations. A graphical representation of mean convergence times for the aligned and unaligned cases are shown in Figs 2 and 3 respectively.

4 DISCUSSION

4.1 Both Aligned and Unaligned Cases

4.1.1 Influence model hierarchy. Likely the most obvious trend in the results is the overwhelming determinant of performance that is the influence model. Essentially, productive alignment work is only done by the effects of orientation; attraction and repulsion generally disorient a flock member. Thus models are essentially 'easier' whenever the effects of attraction and repulsion are minimized. This is mostly the case in the distinct zonal model, where agents will always try to enter the orientation zone of another agent if they are not already in one, and when they are able to orient, they exclusively devote themselves to this task. In the case of inverse zonal, repulsion and attraction are always present on some level, making it more difficult to form cohesive flocks. Lastly, the Couzin

	Selection-Orientation	Mean / Standard Deviation	
		Aligned	Unaligned
Selection-Orientation Combinations	AB-AOSL	112.7 / 55.4	258.1 / 138.8
	AB-FT	144.0 / 68.2	321.0 / 261.4
	AB-OSL	112.6 / 55.4	271.9 / 215.2
	Center-AOSL	126.5 / 67.6	396.7 /148.6
	Center-FT	156.4 / 107.1	442.0 / 318.6
	Center-OSL	129.3 / 72.8	458.3 / 313.6
	Periphery-AOSL	117.0 / 57.3	249.7 / 148.3
	Periphery-FT	152.1 / 69.0	311.3 / 237.0
	Periphery-OSL	116.3 / 57.0	250.4 / 146.6
	K-Means-AOSL	111.8 / 54.7	258.9 / 128.2
	K-Means-FT	140.5 / 65.4	306.7 / 179.0
	K-Means-OSL	111.8 / 54.7	271.4 / 205.2
Average Selection	AB	123.1 / 59.7	283.7 / 205.1
	Center	137.4 / 82.5	459.0 / 343.0
	K-Means	121.4 / 58.3	279.0 / 170.8
	Periphery	128.5 / 61.1	270.5 / 177.3
Average Orientation	AOSL	117.0 / 58.8	310.8 / 203.0
	FT	148.3 / 96.4	345.3 / 249.0
	OSL	117.5 / 60.0	313.0 / 220.2

Table 4: Distinct zonal influence model convergence time means and standard deviations

	Selection-Orientation	Mean / Standard Deviation	
		Aligned	Unaligned
Selection-Orientation Combinations	AB-AOSL	124.0 / 66.3	780.4 / 623.4
	AB-FT	175.3 / 235.2	835.9 / 605.2
	AB-OSL	124.4 / 71.3	782.4 / 576.5
	Center-AOSL	567.2 / 969.8	1606.5 / 1426.8
	Center-FT	637.9 / 907.0	1569.7 / 1235.1
	Center-OSL	547.3 / 832.3	1580.6 / 1205.8
	K-Means-AOSL	113.7 / 64.7	826.3 / 634.3
	K-Means-FT	156.7 / 176.3	878.0 / 732.0
	K-Means-OSL	113.8 / 65.4	880.4 / 811.9
	Periphery-AOSL	161.3 / 103.0	684.4 / 477.3
	Periphery-FT	303.9 / 299.8	718.2 / 515.2
	Periphery-OSL	158.6 / 96.2	698.3 / 618.6
Average Selection	AB	141.2 / 124.3	799.6 / 601.7
	Center	584.1 / 903.1	1585.6 / 1289.2
	K-Means	128.1 / 102.1	861.6 / 726.1
	Periphery	208.0 / 166.3	700.3 / 537.0
Average Orientation	AOSL	241.6 / 301.0	974.4 / 790.5
	FT	318.4 / 404.6	1000.5 / 771.9
	OSL	236.0 / 266.3	985.4 / 803.2

Table 5: Inverse zonal influence model convergence time means and standard deviations

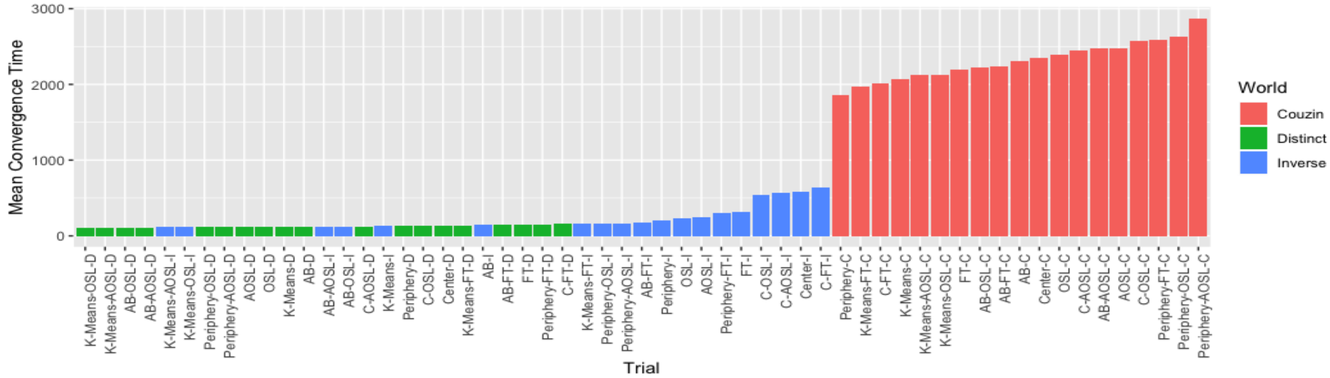

Figure 2: Average convergence times for every combination of leader selection and orientation strategies, including the average convergence time of each individual strategy in trials where all starting agents were nearly aligned.

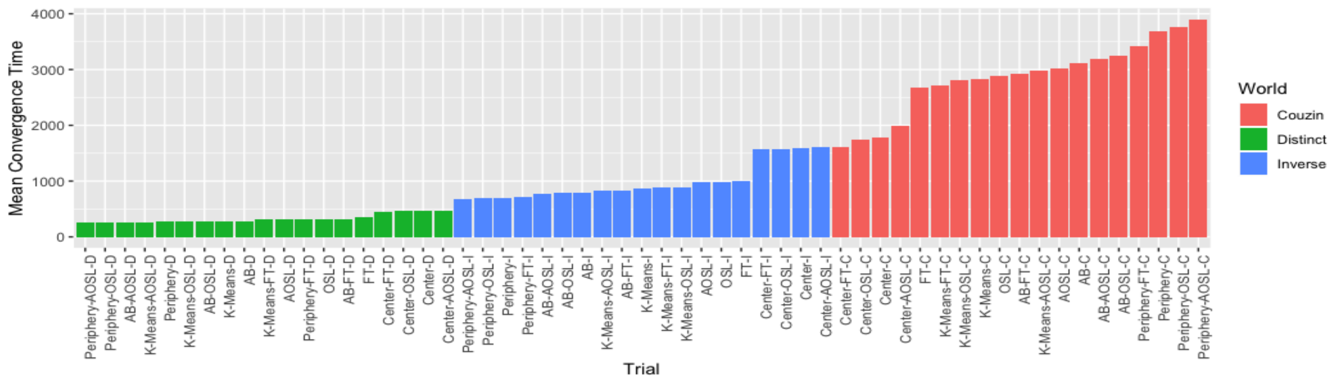

Figure 3: Average convergence times for every combination of leader selection and orientation strategies, including the average convergence time of each individual strategy.

et al. model always forces an agent to devote itself largely to attraction or repulsion, and orientation is rarely given full control. The flock tends to collapse upon itself in this model and is very difficult to steer. Although these effects largely negate the occurrences of isolation, it makes a turn very difficult to accomplish.

4.1.2 Adaptive orientation algorithms suffer with center selection. Adaptive orientation algorithms AOSL and OSL use predictive models to guide their choices. These predictive models rely on assumptions which are not necessarily correct, and which have a greater magnitude of error especially when center selection is used. Pertinent assumption 1 is shared by both algorithms: other influencer agents will adopt the same choice as the one being considered in the algorithm. There is a reason to believe this so long as the neighborhoods of the two influencer agents are similar. Assumption 2 belongs to OSL only, which is precisely the assumption eliminated by AOSL: influencer agents will be influenced in a similar manner as follower agents, and it is equally useful to influencer agents as follower agents. AOSL solves this by skipping over other influencer agents when trying to minimize error, as their alignment does not determine convergence, and they do not listen like follower agents. Both these assumptions are used more often when there is a high amount of influencer agents in an influencer agent's zone of orientation. This is, of course, the case in the center placement strategy, which initially places leaders in a dense circle. As assumptions are frequently violated, the performance of the adaptive algorithms suffer and the simpler non-adaptive algorithm FT performs better than usual only by comparison.

4.1.3 Adaptive orientation algorithms are strong in distinct zonal, are weakened in inverse zonal, and are weakest in Couzin et al. zonal. This is also an issue of assumption within the predictive models of AOSL and OSL. In this case, the assumption made by both is that an agent's zone of orientation is the only contributor to determining its new velocity. This is exactly the case in the distinct zonal model, at least when no agent is in its zone of repulsion, making it a valid assumption in this case, thus not harming performance significantly. However in the case of Couzin et al. zonal, and less so for inverse zonal, this assumption is violated, as section 4.1.1 explains how orientation is rarely the only determinant for velocity updates in these models, and this effect is more pronounced in the former case. The violation of assumptions necessarily leads to a decrease in performance, which is seen in the results.

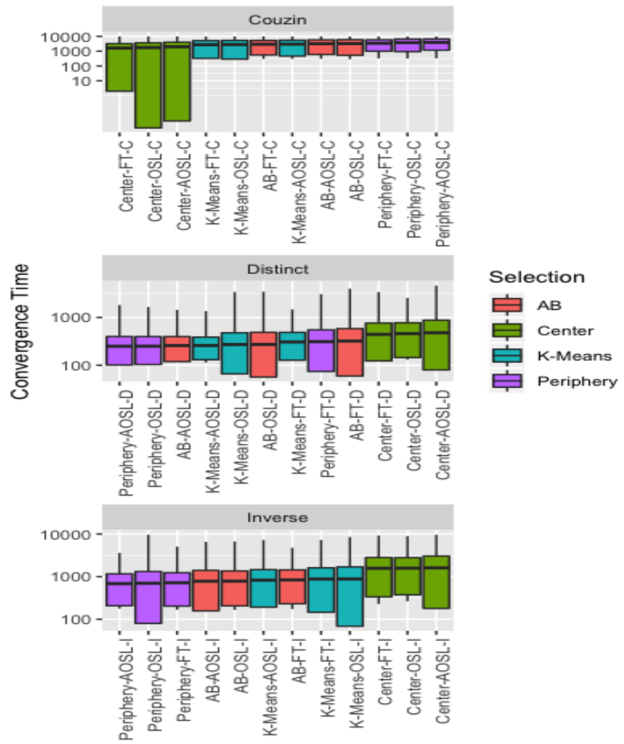

Figure 4: Average convergence time for different selection strategies for each world, including the standard deviation and min and max values on a logarithmic y axis.

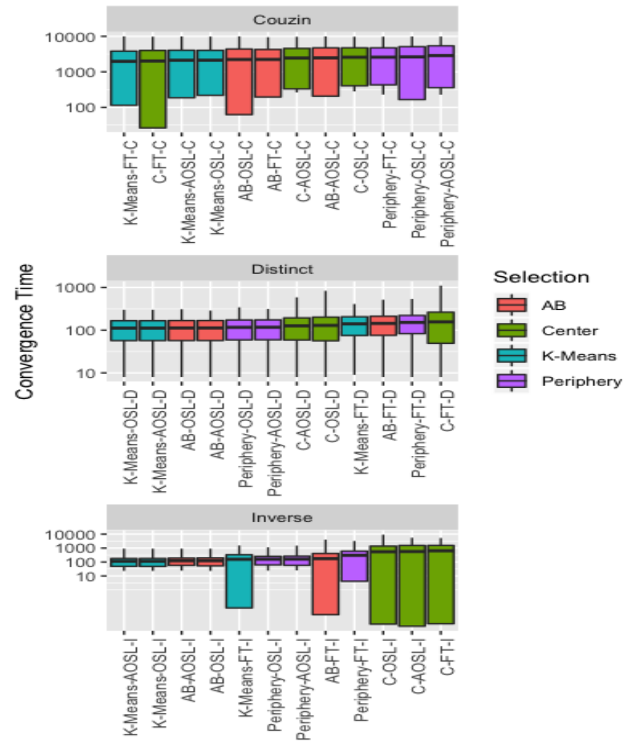

Figure 5: Average convergence time for different selection strategies for each world with agents starting nearly aligned: including standard deviation and min and max values with a logarithmic y axis.

	Selection-Orientation	Mean / Standard Deviation	
		Aligned	Unaligned
Selection-Orientation Combinations	AB-AOSL	2471.8 / 2264.9	3182.7 / 2588.7
	AB-FT	2232.5 / 2036.9	2925.0 / 2372.1
	AB-OSL	2227.8 / 2165.4	3244.0 / 2717.2
	Center-AOSL	2448.8 / 2118.1	1986.9 / 2038.4
	Center-FT	2019.5 / 1993.1	1616.7 / 1614.7
	Center-OSL	2571.0 / 2165.6	1748.3 / 1899.4
	K-Means-AOSL	2120.5 / 1934.1	2974.4 / 2509.8
	K-Means-FT	1972.9 / 1858.4	2720.1 / 2392.5
	K-Means-OSL	2125.7 / 1907.5	2807.7 / 2515.8
	Periphery-AOSL	2874.6 / 2516.3	3903.7 / 2751.7
	Periphery-FT	2581.4 / 2150.5	3416.6 / 2425.5
	Periphery-OSL	2632.6 / 2467.2	3760.3 / 2827.1
Average Selection	AB	2310.7 / 2155.7	3117.2 / 2559.3
	Center	2346.4 / 2092.3	1784.0 / 1850.8
	K-Means	2073.0 / 1900.0	2834.1 / 2472.7
	Periphery	1861.5 / 1576.0	3693.5 / 2668.1
Average Orientation	AOSL	2478.9 / 2208.4	3011.9 / 2472.2
	FT	2201.6 / 2009.7	2669.6 / 2201.2
	OSL	2389.3 / 2176.4	2890.1 / 2489.9

Table 6: Couzin et al. zonal influence model convergence time means and standard deviations

4.2 Unaligned Case

4.2.1 General AOSL dominance over OSL. AOSL makes a general improvement over OSL: it ignores trying to correct the orientation of other influencer agents and focuses on followers. Thus in principle, there should be a general pattern of improvement when comparing the performance of AOSL against OSL, and this is true, but not without exception. Exceptions found within the unaligned case include instances with high concentrations of influencer agents (center selection and the Couzin et al. influence model). Strangely, these are instances where we would expect the highest improvement over OSL, as there is a higher concentration of influencer agents to filter out so that follower agents can be prioritized. Perhaps the removal of the attempt to influence other influencer agents provides much less gain than the loss caused by the greater effect of the faulty assumption that is more pronounced in AOSL in this case: that other influencer agents acting on neighbors will adopt the same velocity choice. The trends between AOSL and OSL within the aligned case are less pronounced and muddled with exceptions that follow no discernible pattern.

4.2.2 Periphery placement is exceptional in inverse zonal. More than any other physical model, inverse zonal suffered from fragmentation of the flock during experimentation. Fragmentation is essentially when the flock splits into two or more distinct groups until eventually reforming, adding a sizable delay in convergence. The periphery selection algorithm clearly places influencer agents in vantage positions to contain possible wayward follower agents, thus it performs the very best in cases of high fragmentation risk (inverse zonal). Knowing this, it would seem center selection would be expected to be terrible in the inverse zonal model (an expectation that is met).

4.2.3 Influencer orientation strategies hardly affect performance in inverse zonal. Mostly this is an illusion caused by balancing effects and the relative importance of selection compared to orientation strategies. As explained in Section 4.1.3, adaptive orientation algorithms suffer slightly from the issues caused by their assumptions, but still have the advantages of making intelligent choices, unlike non-adaptive approaches. AOSL and OSL are similar algorithms, and their performances being only slightly different is no surprise. Lastly, influencer selection is very important particularly in the case of inverse zonal, as described in section 4.2.2, thus appearing to minimize the impact of influencer orientation strategies by comparison.

4.2.4 Center placement performs well in Couzin et. al zonal, but poorly in distinct and inverse zonal. Fragmentation of the flock is occasionally an issue for distinct zonal and especially inverse zonal but not for Couzin et al. zonal as attraction is given such a large sway even initially when agents are already very close. Thus center performs poorly in mitigating fragmentation risk as influencer agents have little control over agents at the periphery which are bound to splinter from the bulk, unlike other placement methods that place some agents specifically on the periphery or at least near it. Fragmentation is the last issue for Couzin et al. zonal. On the contrary, excessive cohesion is the problem. Influencer agents are often separated from the flock as no algorithm considers the importance of staying inside of the flock by making incremental changes to a flock which, in this case, has considerable inertia. Instead, they try to rapidly turn the flock through extreme movement. Influencer agents then rely on chance to place themselves within the flock again, delaying orientation considerably. Center selection, by surrounding the influencer agents with follower agents, makes this separation less common through the boundary created by the repulsive effects of the follower agent composed outer layer.

4.3 Aligned Case

4.3.1 Aligned is slower than unaligned in the Couzin et al. model with center selection. The trend of aligned being faster is intuitively obvious, as there would be less initial fracturing, lost agents, less repelling agents, and therefore full efforts put toward orientation. As well, at times, the flock may be near instantly aligned by chance that the target orientation and initial orientation of the flock are very close. However, a general problem with Couzin et al. zonal is that when the flock is already aligned in the wrong direction, it is incredibly difficult to reorient them back to the desired direction. It is easiest to turn this style of flock in the starting stages of collapse, a phenomenon unique to the Couzin et al. zonal model when all agents head toward the flock's center of mass due to the ever-present force of attraction. At this point, the flock is not yet incorrectly aligned, and it is most susceptible to the guidance of influencer agents. Additionally, only center selection has all of the influencer agents positioned properly to take advantage of this initial absence of inertia during the collapse, as the influencer agents are placed directly on the center of collapse, which again, is the center of mass of the flock. This gives us our unexpected exception: that under specific circumstances (namely center selection and Couzin et al. zonal), initial alignment is a detriment to convergence to the proper alignment.

4.3.2 Perimeter selection strategy performs poorly. Fragmentation is not an issue in the aligned case, so the advantages provided by the perimeter selection strategy (described in 4.2.2) on this front are negated in the cases of inverse and distinct zonal. In the case of Couzin et al. zonal, the unique initial effect of flock collapse leaves influencer agents selected on the edges of the initial flock completely separated from the flock, causing poor performance. Additionally, perimeter selection concentrates the influencer agents in specific areas, limiting their influence over the entire flock. So for every influence model, there are reasons that perimeter selection either has its advantages negated or a new problem created, causing its poor performance overall when agents are initially aligned.

5 CONCLUSIONS AND FUTURE WORK

We experimented with different approaches for effective flock control. We tried all possible combinations of two previously used and one novel influencer selection schemes and two existing and two novel orientation schemes in two existing and one new world (influence models) for both initially aligned and unaligned flocks. Performance is measured in terms of the time steps taken for the flock to converge to the desired orientation. Many interesting and subtle takeaways resulted from the complexity of the interactions between these three aspects, alongside other generalizations that can be made across individual combinations. Notable interactions include the importance of peripheral placement in unaligned initial conditions, the overall dominance of adaptive orientation algorithms, and the incompatibility of adaptive orientation algorithms and center placement (due to high influencer agent concentration). These are important and complex implications for the efficacy of flock control that we have not seen investigated within the existing literature, which often assume a single and simple physical world, and only a single orientation or placement strategy. Physical worlds played a dominant role in determining the effectiveness of approaches, leading us to the conclusion that the study of flock control relies entirely on the accuracy of the physical model which simulates the flock. It is therefore more urgent that more study is put forth into the physical models often used in flock control, as its inaccuracy risks the efficacy of the field of study. In harmony with this conclusion, it was observed that adaptive orientation algorithms thrived to the degree that their predictive models reflected the reality of the model, with simpler methods being preferred for more complicated physical worlds.

The next step of this research would be to characterize abstract features that would be predictive of successful leader placement and orientation strategies. Eventually, the forgiving looping nature of the physical world would have to be eliminated, such that influencer agent strategies would be forced to be developed that are resistant to flock separation.

REFERENCES

[1] Lyuba Alboul, Joan Saez-Pons, and Jacques Penders. 2008. Mixed human-robot team navigation in the guardians project. In *Safety, Security and Rescue Robotics, 2008. SSRR 2008. IEEE International Workshop on*. IEEE, 95–101.

[2] Charles R Brown and Mary Bomberger Brown. 1987. Group-living in cliff swallows as an advantage in avoiding predators. *Behavioral Ecology and Sociobiology* 21, 2 (1987), 97–107.

[3] Iain D Couzin, Jens Krause, Richard James, Graeme D Ruxton, and Nigel R Franks. 2002. Collective memory and spatial sorting in animal groups. *Journal of theoretical biology* 218, 1 (2002), 1–11.

[4] Gaurav Dhiman and Vijay Kumar. 2018. Emperor Penguin Optimizer: A Bio-inspired Algorithm for Engineering Problems. *Knowledge-Based Systems* (2018).

[5] Marco Dorigo and Mauro Birattari. 2011. Ant colony optimization. In *Encyclopedia of machine learning*. Springer, 36–39.

[6] Daniel Y. Fu, Emily S. Wang, Peter M. Kraft, and Barbara J. Grosz. 2018. Influencing Flock Formation in Low-Density Settings. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-1)*. 1604–1612.

[7] Katie Genter and Peter Stone. 2016. Ad hoc teamwork behaviors for influencing a flock. *Acta Polytechnica Journal* 56, 1 (2016).

[8] Katie Genter and Peter Stone. 2016. Adding Influencing Agents to a Flock. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-16)*. 615–623.

[9] Katie Genter, Shun Zhang, and Peter Stone. 2015. Determining Placements of Influencing Agents in a Flock. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS-15)*. 247–255.

[10] Jing Han, Ming Li, and Lei Guo. 2006. Soft control on collective behavior of a group of autonomous agents by a shill agent. *Journal of Systems Science and Complexity* 19, 1 (2006), 54–62.

[11] Xiaofeng Han, Louis F Rossi, and Chien-Chung Shen. 2007. Autonomous navigation of wireless robot swarms with covert leaders. In *Proceedings of the 1st international conference on Robot communication and coordination*. IEEE Press, 27.

[12] J Herskin and JF Steffensen. 1998. Energy savings in sea bass swimming in a school: measurements of tail beat frequency and oxygen consumption at different swimming speeds. *Journal of Fish Biology* 53, 2 (1998), 366–376.

[13] Benedict G Hogan, Hanno Hildenbrandt, Nicholas E Scott-Samuel, Innes C Cuthill, and Charlotte K Hemelrijk. 2017. The confusion effect when attacking simulated three-dimensional starling flocks. *Royal Society open science* 4, 1 (2017), 160564.

[14] Kathryn A Hoppe. 2004. Late Pleistocene mammoth herd structure, migration patterns, and Clovis hunting strategies inferred from isotopic analyses of multiple death assemblages. *Paleobiology* 30, 1 (2004), 129–145.

[15] Ali Jadbabaie, Jie Lin, and A Stephen Morse. 2003. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on automatic control* 48, 6 (2003), 988–1001.

[16] Shin-Young Jung, Daniel S Brown, and Michael A Goodrich. 2013. Shaping couzin-like torus swarms through coordinated mediation. In *Systems, man, and cybernetics (smc), 2013 ieee international conference on*. IEEE, 1834–1839.

[17] Dervis Karaboga and Bahriye Basturk. 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of global optimization* 39, 3 (2007), 459–471.

[18] James Kennedy. 2011. Particle swarm optimization. In *Encyclopedia of machine learning*. Springer, 760–766.

[19] Andreas Kolling, Steven Nunnally, and Michael Lewis. 2012. Towards human control of robot swarms. In *Proceedings of the seventh annual ACM/IEEE international conference on human-robot interaction*. ACM, 89–96.

[20] Céline Le Bohec, Michel Gauthier-Clerc, and Yvon Le Maho. 2005. The adaptive significance of crèches in the king penguin. *Animal Behaviour* 70, 3 (2005), 527–538.

[21] David MacKay. 2003. An Example Inference Task: Clustering. In *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 284–292.

[22] Stefano Marras, Shaun S Killen, Jan Lindström, David J McKenzie, John F Steffensen, and Paolo Domenici. 2015. Fish swimming in schools save energy regardless of their spatial position. *Behavioral ecology and sociobiology* 69, 2 (2015), 219–226.

[23] Reza Olfati-Saber. 2006. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on automatic control* 51, 3 (2006), 401–420.

[24] GVN Powell. 1974. Experimental analysis of the social value of flocking by starlings (Sturnus vulgaris) in relation to predation and foraging. *Animal Behaviour* 22, 2 (1974), 501–505.

[25] Craig W Reynolds. 1987. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH computer graphics*, Vol. 21. ACM, 25–34.

[26] Shahrzad Saremi, Seyed Mohammad Mirjalili, and Seyedali Mirjalili. 2014. Chaotic krill herd optimization algorithm. *Procedia Technology* 12 (2014), 180–185.

[27] Kathryn J Schneider. 1984. Dominance, predation, and optimal foraging in white-throated sparrow flocks. *Ecology* 65, 6 (1984), 1820–1827.

[28] Joan B Silk. 2007. Social components of fitness in primate groups. *Science* 317, 5843 (2007), 1347–1351.

[29] Housheng Su, Xiaofan Wang, and Zongli Lin. 2009. Flocking of multi-agents with a virtual leader. *IEEE Trans. Automat. Control* 54, 2 (2009), 293–307.

[30] Rohan Tiwari, Puneet Jain, Sachit Butail, Sujit P. Baliyarasimhuni, and Michael A. Goodrich. 2017. Effect of Leader Placement on Robotic Swarm Control. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS '17)*. 1387–1394.

[31] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. 1995. Novel type of phase transition in a system of self-driven particles. *Physical review letters* 75, 6 (1995), 1226.

[32] Phillip Walker, Saman Amirpour Amraii, Michael Lewis, Nilanjan Chakraborty, and Katia Sycara. 2014. Control of swarms with multiple leader agents. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*. IEEE, 3567–3572.

[33] Justin Werfel and Radhika Nagpal. 2006. Extended stigmergy in collective construction. *IEEE Intelligent Systems* 21, 2 (2006), 20–28.

[34] David Wilson. 2009. Causes and benefits of chick aggregations in penguins. *The Auk* 126, 3 (2009), 688–693.