


ON THE EFFECTS OF COLLISION AVOIDANCE  
ON EMERGENT SWARM BEHAVIOR

by

Christopher ArieH Taylor  
A Thesis  
Submitted to the  
Graduate Faculty  
of  
George Mason University  
In Partial fulfillment of  
The Requirements for the Degree  
of  
Master of Science  
Electrical Engineering

Committee:

|  |   |
|--|---|
| <br>_____ | Dr. Cameron Nowzari, Thesis Director                      |
| <i>zhangfeitian</i><br>_____   | Dr. Feitian Zhang, Committee Member                       |
| <i>Abbas Kazim</i><br>_____  | Dr. Abbas Zaidi, Committee Member                         |
| <i>Monson Hayes</i><br>_____   | Dr. Monson Hayes, Department Chair                        |
| <i>Kenneth Ball</i><br>_____   | Dr. Kenneth Ball, Dean, Volgenau<br>School of Engineering |

Date: May 17, 2020

Summer Semester 2020  
George Mason University  
Fairfax, VA

On the Effects of Collision Avoidance on Emergent Swarm Behavior

A thesis submitted in partial fulfillment of the requirements for the degree of  
Master of Science at George Mason University

By

Christopher Arieh Taylor  
Bachelor of Science  
University of California, Santa Barbara, 2014

Director: Dr. Cameron Nowzari, Professor  
Department of Electrical and Computer Engineering

Summer Semester 2020  
George Mason University  
Fairfax, VA

Copyright © 2020 by Christopher Arieh Taylor  
All Rights Reserved

## Acknowledgments

This work was supported by the Department of the Navy, Office of Naval Research (ONR), under federal grant N00014-19-1-2121.

# Table of Contents

|   | Page |
|---|------|
| List of Tables . . . . .                                  | v    |
| List of Figures . . . . .                                 | vi   |
| Abstract . . . . .  | vii  |
| 1 Introduction . . . . .                                  | 1    |
| 1.1 Background . . . . .                                  | 1    |
| 1.2 Problem Formulation . . . . .                         | 3    |
| 1.2.1 Individual Agent Model . . . . .                    | 3    |
| 1.2.2 Desired Global Behavior: Ring State . . . . .       | 4    |
| 2 Methodology . . . . .                                   | 7    |
| 2.1 Collision Avoidance . . . . .                         | 7    |
| 2.2 Measuring Emergent Behavior Quality . . . . .         | 9    |
| 2.3 Finding Safe Collision Avoidance Parameters . . . . . | 11   |
| 3 Results . . . . .                                       | 16   |
| 3.1 Comparing Collision Avoidance Algorithms . . . . .    | 16   |
| 3.2 Choosing Collision Avoidance Parameters . . . . .     | 16   |
| 3.3 Conclusion . . . . .                                  | 18   |
| Bibliography . . . . .                                    | 20   |

## List of Tables

Table

Page

## List of Figures

| Figure | Page   |    |
|--------|--|----|
| 1.1    | Positions and velocities of 12 agents for different parameters of $\mathbf{f}_c$ . The other parameters are held constant at $\alpha = 0.001, t_d = 2.5, \beta = 1, v_0 = 0.12, \ell_r = 1, a_{\max} = 0.6, r = 0.4$ . Also shown for each is the orderliness metric $\lambda$ . . . . . | 6  |
| 3.1    | A comparison how each collision avoidance strategy scales with the number of agents $N$ . Left: the convergence quality $\lambda$ . Right: the crash rate. . . . .   | 17 |
| 3.2    | Left: The convergence quality $\lambda$ . Right: the crash rate as a function of $r$ and $c_r$ . The white lines show safe values of $c_r$ as a function of $r$ , where safety is defined by $h_2(\mathbf{p}) = 0$ and $h_3(\mathbf{p}) = 0$ . . . . .                                   | 17 |
| 3.3    | A similar plot to Fig. 3.2 except $c_r = 0.2$ and $\ell_r$ is varied. . . . .  | 18 |

# Abstract

ON THE EFFECTS OF COLLISION AVOIDANCE ON EMERGENT SWARM BEHAVIOR

Christopher Arie Taylor

George Mason University, 2020

Thesis Director: Dr. Cameron Nowzari

Swarms of autonomous agents, through their decentralized and robust nature, show great promise as a future solution to the myriad missions of business, military, and humanitarian relief. Swarms can be useful purely as a theoretical abstraction or in simulation, but in many applications the swarm needs to be deployed on actual hardware platforms. The diverse nature of mission sets creates the need for a variety of hardware platforms, each with their own capabilities and limitations, for instance with sensing, actuation, communications, environmental disturbances, and structural robustness. In particular, the structural robustness of the platform, or lack thereof, seems to have a great effect on the viability of swarming behaviors where collisions might be an issue. Certain swarm behaviors have been demonstrated on platforms where collisions between agents are harmless, but on many platforms collisions are prohibited since they would damage the agents involved. The available literature typically assumes that collisions can be avoided by adding a collision avoidance algorithm on top of an existing swarm behavior. Through an illustrative example in our experience replicating a particular behavior, we show that this can be difficult to achieve since the swarm behavior can be disrupted by the collision avoidance.



Furthermore, if collisions cause irreparable damage to the agents involved, we show that weakening the collision avoidance can *also* disrupt the intended swarm behavior, since destroyed agents are no longer able to interact with the rest of the swarm and their sudden disappearance dramatically alters the behavior of nearby agents. We introduce metrics quantifying the level of disruption in our swarm behavior and propose a technique that is able to assist in tuning the collision avoidance algorithm such that the goal behavior is achieved as best as possible while collisions are avoided. We validate our results through simulation.

# Chapter 1: Introduction

## 1.1 Background

Swarms have been extensively studied and are an attractive choice for many applications due to their decentralized nature and robustness against individual failures [1–3]. A common goal with decentralized control in swarms is to achieve an *emergent behavior* [1, 4], where the collective behavior of the swarm has properties that the behaviors of individual agents lack. This is desirable when agents lack global awareness of the higher level goal or other agents, often due to limited sensing or computation capabilities. A common goal is to be able to replicate these emergent behaviors on physical platforms in the real world.

In applications such as computer graphics [1, 5] or optimization algorithms [6], swarms are purely an abstraction and we are not concerned with the physical platforms they run upon. Yet, for other new-age swarming applications, we would like to leverage these existing behaviors validated in simulation and bring them into the real world. However, physical platforms come with many new limitations, among them actuation constraints and their use of physical space. For instance, in a dense swarm of ground robots, agents need to coordinate their actions to avoid overcrowding or deadlocks [7]. With aerial platforms, special care needs to be taken in the agents’ construction to ensure they can collide without damage [8].

Unless overcrowding becomes a problem, these platforms not as affected by physical space constraints because they are relatively small and slow and can collide harmlessly. With larger, faster platforms such as the quadrotors used in DARPA’s OFFSET program [9], this is not the case and collisions would likely severely damage the agents involved. Yet, regardless of whether the platforms collide harmlessly or catastrophically, in many existing works these physical limitations are ignored and agents are allowed to simply pass right

through one another. We suspect many existing works created in simulation could have serious issues if we attempt to replicate them on such platforms. In particular, it seems very few works have paid any attention to simulating platforms where collisions are catastrophic.

*Literature review:* One might think this problem could be ignored by deploying a collision-avoidance algorithm. Some simple approaches to collision avoidance include using artificial potential fields where agents “repel” each other [10] or “gyroscopic” forces where agents steer around each other and do not change their speed [11]. Later works take a more rigorous approach and introduce more theoretically sound *minimally invasive* controllers such as *optimal reciprocal collision avoidance* (ORCA) [12] and *control barrier certificates* [13]. In both cases, agents have a primary goal in mind and select a ‘minimally invasive’ control input that will avoid collisions in a way that stays as true as possible to the intended behavior. To validate their techniques, most works use specially constructed test scenarios [11–13], usually involving a group of agents starting on a circle and heading to the point directly opposite on the circle. Unfortunately, most of these works do not investigate the effect of the collision avoidance algorithms on the original intended behavior of the swarm.

A few works combine the study of emergent swarm behavior with collision avoidance, such as in search and coverage control problems [14, 15], flocking [16], or formation control [17]. However, in these examples, the intended behaviors already keep agents away from each other so adding collision avoidance does not seem prohibitive.

Instead, in this work we consider a behavior where the intended behavior of the swarm is less aligned with avoiding collisions. Examples of such behaviors include *milling* [18], where agents orbit a common point in a dense group, and *double milling* [4, 10], where agents rotate in opposing directions, frequently encountering each other at high relative speeds. We believe more work is needed to understand how behaviors like these interact with collision avoidance.

*Statement of contributions:* In this work we rigorously study the effects of different collision avoidance strategies on a particular swarming algorithm proposed by Szwaykowska et al. [10]. Under a particular set of parameters, the double-milling behavior is shown to

emerge among the swarm of agents when collisions among agents are not modeled. We then impose two physical constraints on the system (no collisions and limited acceleration) and study how we can still achieve the desired behavior.

Specifically, we first introduce a metric that captures how well the agents perform the desired milling behavior. Using this metric, we explore two different collision avoidance techniques (potential fields and gyroscopic forces) under a very large set of parameters to quantitatively understand how active collision avoidance disrupts the intended behavior of the swarm. We also explore two more sophisticated collision avoidance strategies, ORCA [12] and control barrier certificates [13], which can be found in a followup paper [19] due to space constraints. Finally, given a particular choice of a collision avoidance strategy, we show how to tune the parameters of the algorithm to ensure collisions among agents are avoided while preserving the intended behavior as much as possible. Our results suggest that a successful algorithm that can guarantee the emergence of the desired behavior and no collisions simultaneously should be co-designed rather than combining existing swarming algorithms with existing collision avoidance strategies.

## 1.2 Problem Formulation

This paper is concerned with the deployment of physical swarm systems in which collisions between agents are catastrophic. In such scenarios, we are interested in understanding the effect that this added constraint has on the ability of the swarm to reach an intended globally emergent behavior. More specifically, we aim to understand the effects that various collision avoidance algorithms have when combined with a particular swarming algorithm.

### 1.2.1 Individual Agent Model

We want to deploy swarms on physical swarm systems, so we focus on simple agent models that capture the physical aspects we are most concerned with. Letting  $\mathbf{r}_i \in \mathbb{R}^2$  be the position of agent  $i \in \{1, \dots, N\}$  in a swarm of  $N$  agents, we consider double-integrator dynamics  $\ddot{\mathbf{r}}_i(t) = u_i(t)$  with the following two constraints at all times  $t \in \mathbb{R}_{\geq 0}$ .

**C1. Limited acceleration.** Since we cannot guarantee that our desired dynamics obeys an acceleration limit, we “clip” the acceleration based on a limit  $a_{\max}$  as

$$\ddot{\mathbf{r}}_i = \mathbf{clip}(\mathbf{u}_i, a_{\max}), \quad (1.1)$$

where  $\mathbf{clip}(\cdot)$  caps the acceleration in the direction of  $\mathbf{u}_i^*$ ,

$$\mathbf{clip}(\mathbf{x}, a) = \begin{cases} \mathbf{x} & \|\mathbf{x}\| < a, \\ a \frac{\mathbf{x}}{\|\mathbf{x}\|} & \text{otherwise.} \end{cases} \quad (1.2)$$

**C2. No collisions.** Letting  $r > 0$  represent the our radius of each agent, agents must obey

$$\|\mathbf{r}_i(t) - \mathbf{r}_j(t)\| > 2r, \quad (1.3)$$

for all  $i, j \in \{1, \dots, N\}$ . Without constraint C1, agents can increase their acceleration arbitrarily large which is impractical for swarms operating with physical limitations in the real world. If C2 is violated, we consider the agents “destroyed” and respawn them in a safe location in order to keep the number of agents  $N$  fixed; see [19] for more details.

### 1.2.2 Desired Global Behavior: Ring State

Given the model above, we now introduce our specific desired behavior for the swarm system. The dynamics formulation in [10] is capable of producing a few behaviors, but we want the “ring state”, also known as double-milling, where agents self-organize to orbit around a common point. Given the model in Eq. (1.1), the controller we replicate from [10]

is

$$\begin{aligned} \mathbf{u}_i &= \beta(v_0^2 - \|\dot{\mathbf{r}}_i\|^2)\dot{\mathbf{r}}_i + \mathbf{f}_c(\mathbf{r}_i, \dot{\mathbf{r}}_i, \mathcal{N}_i) \\ &+ \frac{\alpha}{N-1} \sum_{j \neq i} (\mathbf{r}_j(t-t_d) - \mathbf{r}_i(t)), \end{aligned} \tag{1.4}$$

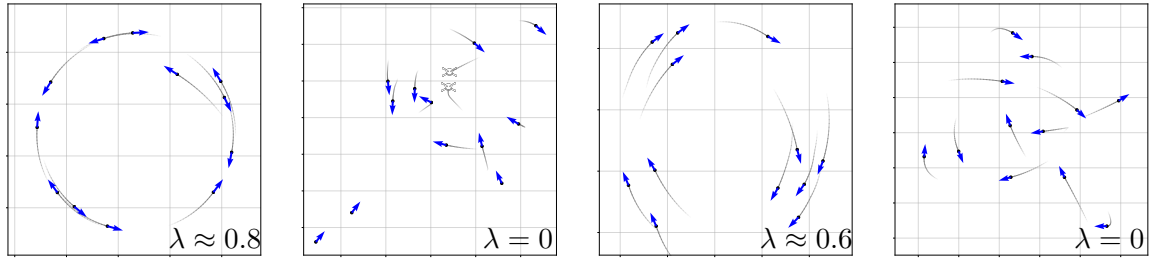
where  $\mathbf{u}_i^*$  is comprised of three terms (in order): keep the agent's speed at approximately  $v_0 > 0$  with gain  $\beta > 0$ , avoid nearby agents using a collision avoidance term  $\mathbf{f}_c$ , and attract toward the delayed position of other agents  $t_d$  seconds in the past, where the strength of the attraction is weighted by  $\alpha$ . The original formation in [10] uses a fixed communication graph for the delayed channel and assumes agents can only sense a subset of their neighbors, but in this work we assume all-to-all communication is available to help the ring state emerge as easily as possible, as our goal is to understand the effects that collision avoidance has on the ideal behavior.

The neighbor set  $\mathcal{N}_i$  is agent  $i$ 's *local* neighbors and is based on a circular sensing area with radius  $\ell_r$  defined by

$$\begin{aligned} \mathcal{N}_i &= \{(\mathbf{r}_j, \dot{\mathbf{r}}_j) \mid j \in \{1, \dots, N\} \setminus \{i\}, \\ &\|\mathbf{r}_i - \mathbf{r}_j\| \leq \ell_r\}. \end{aligned} \tag{1.5}$$

It is important to note how these are used in Eq. (1.4): if an agent needs the state of one of its local neighbors in  $\mathcal{N}_i$  it is received with no delay, but sensing far-away neighbors incurs a delay of  $t_d$  seconds. This simulates a situation where agents can see their immediate neighbors and receive information on far-away agents through a separate channel.

If we select suitable parameters for  $t_d, v_0, \beta, \alpha$  and ignore the collision avoidance term  $\mathbf{f}_c$  and constraint C2, a rotating ring emerges as shown in figure Fig. 1.1a. By ignoring C2, the agents simply move through one another and the desired swarm behavior emerges without problems.



(a) Collisions and collision avoidance disabled.

(b) Collisions and collision avoidance enabled and  $c_r = 0.15$  is too low.

(c) Collisions and collision avoidance enabled and  $c_r = 0.3$  is sufficient.

(d) Collisions and collision avoidance enabled but  $c_r = 4$  is too strong.

Figure 1.1: Positions and velocities of 12 agents for different parameters of  $\mathbf{f}_c$ . The other parameters are held constant at  $\alpha = 0.001$ ,  $t_d = 2.5$ ,  $\beta = 1$ ,  $v_0 = 0.12$ ,  $\ell_r = 1$ ,  $a_{\max} = 0.6$ ,  $r = 0.4$ . Also shown for each is the orderliness metric  $\lambda$ .

However, since we are interested in ensuring collisions do not occur, we must consider various collision avoidance strategies  $\mathbf{f}_c$  and their effect on the intended behavior.

## Chapter 2: Methodology

With our problem defined, we must first choose different types of collision avoidance strategies  $\mathbf{f}_c$  for the agents to employ. In trying to understand the effects these have on the desired emergent behavior, we note that we are essentially attempting to capture a qualitative property. Figure 1.1 clearly shows that the question of whether the intended behavior successfully emerged has a non-binary answer. Thus, after discussing different collision avoidance mechanisms in Section 2.1, we propose a quantifiable metric in Section 2.2 to enable comparison between various states to determine which produces the desired emergent behavior ‘better’. Finally, we utilize these tools in Section 2.3 to investigate how well collision avoidance strategies can be tuned to achieve the desired emergent behavior while satisfying physical constraints C1 and C2.

### 2.1 Collision Avoidance

We compare two collision avoidance schemes to be used for the anti-crash force  $\mathbf{f}_c$  in Eq. (1.4). The first choice is a potential-fields scheme presented with [10], based on the gradient of the potential function

$$\mathbf{f}_p(\mathbf{r}_i, \dot{\mathbf{r}}_i, \mathcal{N}_i) = \nabla_{\mathbf{r}_i} \sum_{(\mathbf{r}_j, \dot{\mathbf{r}}_j) \in \mathcal{N}_i} c_r \exp\left(-2 \frac{\|\mathbf{r}_i - \mathbf{r}_j\|}{\ell_r}\right). \quad (2.1)$$

The second is a “gyroscopic” force presented in [11]. This produces a force orthogonal to the agents velocity that “steers” the agent without changing its speed. It can be written



in closed-form as

$$\mathbf{f}_g(\mathbf{r}_i, \dot{\mathbf{r}}_i, \mathcal{N}_i) = R_{90^\circ} \frac{\dot{\mathbf{r}}_i}{\|\dot{\mathbf{r}}_i\|} \text{sgnz}((\mathbf{r}_j^* - \mathbf{r}_i) \times \dot{\mathbf{r}}_i) U(\|\mathbf{r}_i - \mathbf{r}_j^*\|), \quad (2.2)$$

where the cross product is the 2D analog, i.e.  $\mathbf{u} \times \mathbf{v} = u_x v_y - u_y v_x$ ,  $R_{90^\circ}$  is a  $90^\circ$  rotation matrix,  $(\mathbf{r}_j^*, \dot{\mathbf{r}}_j^*) = \arg \min_{(\mathbf{r}_j, \dot{\mathbf{r}}_j) \in \mathcal{N}_i} \|\mathbf{r}_i - \mathbf{r}_j\|$  is the *nearest* agent's state, and

$$\text{sgnz}(x) = \begin{cases} 1 & x = 0 \\ \text{sgn}(x) & \text{otherwise} \end{cases} \quad (2.3)$$

is the sign function modified such that the agent is forced to steer left during a perfect head-on collision to prevent a situation where  $\mathbf{f}_g = 0$ . The function  $U(d)$  represents a potential controlling the magnitude of the steering force. As [11] specifies, the magnitude  $U(d)$  is arbitrary so we choose

$$U(d) = 2 \frac{c_r}{\ell_r} \exp\left(-2 \frac{d}{\ell_r}\right) \quad (2.4)$$

such that the force magnitude is exactly the same as the method of potential fields Eq. (2.1) with one other agent.

Fig. 1.1 explores what happens with the potential-fields collision avoidance approach  $\mathbf{f}_c = \mathbf{f}_p$  as we change just the strength parameter  $c_r$  while leaving all others parameters fixed. We are interested in swarming algorithms that are able to guarantee both the emergence of the desired behavior while actively avoiding collisions. Unfortunately, Fig. 1.1 also demonstrates that we are interested in understanding a qualitative property of the entire swarm. This motivates the need to define a more precise metric.

## 2.2 Measuring Emergent Behavior Quality

We measure the quality of the emergent behavior both through the amount of collisions and through specialized metrics to quantify how closely the behavior matches the desired ring formation. Many other works define metrics to quantify emergent behavior, for instance [20] uses *polarity* and *normalized angular momentum*, to quantify a rotating mill similar to our ring state, [21] uses *group polarization* to quantify alignment in fish schooling, and [22] uses a correlation function to quantify alignment in starling flocks. Similar to those in [20], we introduce two metrics to quantify the quality of the emerged behavior: the “fatness”  $\Phi$ , which characterizes how thick the ring is relative to its inner diameter, and “tangentsness”  $\tau$ , the degree to which agents’ velocities are aligned tangentially to the ring.

Formally, letting  $\mu$  be the average position of all agents, i.e.  $\mu = \frac{1}{N} \sum_{i=1}^N \mathbf{r}_i$  and  $r_{\min}$  and  $r_{\max}$  be the minimum and maximum distance of the formation to  $\mu$ , respectively,

$$r_{\min} = \min_{i \in \{1, \dots, N\}} \|\mathbf{r}_i - \mu\|, \quad r_{\max} = \max_{i \in \{1, \dots, N\}} \|\mathbf{r}_i - \mu\|,$$

fatness and tangentsness are defined as

$$\Phi(t) = 1 - \frac{r_{\min}^2(t)}{r_{\max}^2(t)}, \quad \tau(t) = \frac{1}{N} \sum_{i=1}^N \left| \frac{\mathbf{r}_i - \mu}{\|\mathbf{r}_i - \mu\|} \cdot \frac{\dot{\mathbf{r}}_i}{\|\dot{\mathbf{r}}_i\|} \right|. \quad (2.5)$$

In other words, the fatness  $\Phi$  is the proportion of empty space available in the center of the formation, where  $\Phi = 0$  implies a perfectly thin ring and  $\Phi = 1$  implies an entirely filled-in disc. The tangentsness  $\tau$  measures the average cosine of the angle between an agent’s velocity vector and the normal vector to the circle centered at  $\mu$ , where  $\tau = 0$  represents perfect alignment and  $\tau = 1$  represents maximum disorder. The tangentsness is similar to the *normalized angular momentum* measure in [20] except that it ignores each agent’s absolute speed and only considers alignment.

The fatness  $\Phi$  and tangentsness  $\tau$  metrics are defined for one instant in time. Since it is

more useful to consider the behavior of the swarm in steady-state, we define

$$\bar{\Phi}(t) = \frac{1}{T} \int_{t-T}^t \Phi(u) du, \quad \bar{\tau}(t) = \frac{1}{T} \int_{t-T}^t \tau(u) du, \quad (2.6)$$

where  $T$  is the interval over which an average is recorded. We choose  $T = 20,000$  in our tests.

We additionally define a single *orderliness* metric  $\lambda \in [0, 1]$  that combines the steady-state fatness  $\bar{\Phi}$  and steady-state tangentness  $\bar{\tau}$  of the system into one number as  $\lambda = 1 - \max(\bar{\Phi}, \bar{\tau})$ , where  $\lambda = 1$  represents a perfect ring and  $\lambda = 0$  represents maximum disorder. Fig. 1.1 shows the approximate values of  $\lambda$  under each formation.

To quantify crashes, we consider the *crash rate* in collisions per second since we are interested in the steady-state behavior of the swarm independently of how long the swarm has been operational. Since we consider collisions to be catastrophic, it does not make sense to simply count how many times agents violate constraint C2 with no further consequences, which makes more sense for ‘soft’ agents like fish [21]. To capture this, we remove any agents that violate constraint C2 from their current location and ‘respawn’ them at a safe distance away from the rest of the swarm. This is necessary since we are interested in steady-state behavior of the swarm for a *specific* number of agents  $N$  and allowing  $N$  to decrease will lead to unfairly biased analysis.

Equipped with our orderliness metric  $\lambda$  and crash rate metric, we can now study our problem in a more quantitative way. Figs. 3.2 and 3.3 explore the results as we vary the sizes of the agents  $r$ , the collision avoidance gain  $c_r$ , and the sensing range  $\ell_r$ . The white curves are explained in Section 2.3.

Figs. 3.2 and 3.3 seem to suggest that there exists a hard boundary in the parameter space separating a successfully emerged behavior and one that fails due to too many collisions. The edge of this boundary just before agents begin colliding seems to provide the best behavior quality  $\lambda$ . This suggests that if our goal is to utilize swarming algorithms with various collision avoidance strategies, we would like to operate right at this boundary.

We want to find this boundary in a more methodical way than sampling the parameter space.

## 2.3 Finding Safe Collision Avoidance Parameters

Here we want to find the boundary in parameter space that we identify in Section 2.2. Specifically, given all the parameters except one, we wish to find the value of the missing parameter that maximizes the behavior quality  $\lambda$ , which as a side effect minimizes crashes. We conjecture based on Figs. 3.2 and 3.3 that the best parameters are those which are on the *edge of safety*: just barely strong enough to avoid a crash but not too strong to interfere in the behavior. Rather than analyze the entire swarm, we take the myopic view of one agent and identify conditions under which it can guarantee no collisions with a fixed number of other agents.

We consider our test scenario to be a fixed number of agents avoiding each other while in each other’s sensing radii. We find that while agents are closer avoiding each other the anti-crash term  $\mathbf{f}_c$  tends to be much stronger than the other terms in Eq. (1.4). Thus, as an approximation we represent the agent dynamics as

$$\ddot{\mathbf{r}}_i = \mathbf{clip}(\mathbf{f}_c(\mathbf{r}_i, \dot{\mathbf{r}}_i, \mathcal{N}_i), a_{\max}). \quad (2.7)$$

Let  $\mathbf{p}$  be the set of parameters used to define  $\mathbf{f}_c$ ,  $\mathbf{p} = (c_r, \ell_r, a_{\max}, v_0, r)$ . We consider a selection of parameters  $\mathbf{p}$  to be on the edge of safety if the closest distance agents can get under our test scenario is exactly  $2r$ .

### Safety with two agents

For this case, our starting point is any state where two agents have just entered each other’s sensing radii (at a distance  $\ell_r$  away from each other) traveling at a speed up to  $v_0$ . For analysis purposes and with a slight abuse of notation, we redefine the states to be in coordinates relative to agent 1 rather than a fixed frame. We introduce the reachable set

$\mathcal{S}_2(\mathbf{p})$ , the set of all possible relative positions two agents can be in while avoiding a crash using the dynamics of Eq. (2.7) with parameters  $\mathbf{p}$ , which is

$$\begin{aligned} \mathcal{S}_2(\mathbf{p}) = \{ & (\mathbf{r}_1(t), \dot{\mathbf{r}}_1(t), \mathbf{r}_2(t), \dot{\mathbf{r}}_2(t)) \in \mathbb{R}^8 \mid \\ & \mathbf{r}_1(0) = \mathbf{0}, \quad \mathbf{r}_2(0) = (\ell_r, 0), \\ & \|\dot{\mathbf{r}}_2(0)\| \leq v_0, \quad \|\dot{\mathbf{r}}_1(0)\| \leq v_0, \\ & \text{Eq. (2.7) holds, } t \geq 0\}, \end{aligned} \tag{2.8}$$

where  $\ell_r, v_0$  come from the parameters  $\mathbf{p}$ . Note that this is the set of positions relative to the starting state of agent 1, but a rigid transformation applied to all coordinates can transform this scenario into anything where  $\|\mathbf{r}_1 - \mathbf{r}_2\| = \ell_r, \mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^2$ . We define the “headroom”  $h_2(\mathbf{p})$  as the available space agents have in the worst case

$$h_2(\mathbf{p}) = \min_{\mathbf{r}_1, \dot{\mathbf{r}}_1, \mathbf{r}_2, \dot{\mathbf{r}}_2 \in \mathcal{S}_2(\mathbf{p})} \|\mathbf{r}_2 - \mathbf{r}_1\| - 2r. \tag{2.9}$$

Thus, the parameters on the edge of safety that guarantee no collisions with two agents are  $\mathbf{p} = \arg \min_{\mathbf{p}} h_2(\mathbf{p})$  s.t.  $h_2(\mathbf{p}) > 0$ . We conjecture that the solution for  $h_2(\mathbf{p})$  is a head-on collision at full speed, that is, we consider only the subset of the reachable set  $\mathcal{S}_2$  where  $\dot{\mathbf{r}}_1(0) = (v_0, 0)$  and  $\dot{\mathbf{r}}_2(0) = (-v_0, 0)$ . This simplifies finding the solution to Eq. (2.9) without running any optimization routines.

### Safety with three agents

Clearly there will be more than two agents coming in contact with each other so we extend the logic of the previous discussion to three agents. Similar to before, we consider the reachable set for three agents who are avoiding each other using the dynamics of Eq. (2.7). This scenario, specifically, consists of:

1. Two agents come within  $\ell_r$  of one another and begin avoiding each other, i.e., their

states are in  $\mathcal{S}_2$ .

2. A third agent enters at the edge of either of the first two agents' sensing radius.

Using this setup we define  $\mathcal{S}_3$  as

$$\begin{aligned}
\mathcal{S}_3(\mathbf{p}) = & \\
& \{(\mathbf{r}_1(t), \dot{\mathbf{r}}_1(t), \dots, \mathbf{r}_3(t), \dot{\mathbf{r}}_3(t) \in \mathbb{R}^{12} \mid \\
& \mathbf{r}_1(0), \dot{\mathbf{r}}_1(0), \mathbf{r}_2(0), \dot{\mathbf{r}}_2(0) \in \mathcal{S}_2(\mathbf{p}), \\
& \min(\|\mathbf{r}_3(0) - \mathbf{r}_1(0)\|, \|\mathbf{r}_3(0) - \mathbf{r}_2(0)\|) = \ell_r, \\
& \|\dot{\mathbf{r}}_3(0)\| \leq v_0, \|\mathbf{r}_i - \mathbf{r}_j\| \leq \ell_r \forall i, j \in \{1, 2, 3\}, \\
& \text{Eq. (2.7) holds, } t \geq 0\}.
\end{aligned} \tag{2.10}$$

The headroom  $h_3$  for three agents is defined similar to  $h_2$ ,

$$\begin{aligned}
h_3(\mathbf{p}) = \min_{i \neq j} \|\mathbf{r}_i - \mathbf{r}_j\| - 2r \\
\text{subj. to } \mathbf{r}_1, \dot{\mathbf{r}}_1, \dots, \mathbf{r}_3, \dot{\mathbf{r}}_3 \in \mathcal{S}_3(\mathbf{p}).
\end{aligned} \tag{2.11}$$

We find the solution to the three agent case  $h_3$  using two algorithms: simulated annealing [23] and differential evolution [24] included with the Scipy package [25], and verify that they both arrive at the same answer. For all choices of  $\mathbf{p}$  that avoid a crash, the solution we find for the three agent worst case  $h_3$  can be described as

1. Agent 1 “boosts” the speed of agent 2 past  $v_0$
2. Agent 2 undergoes a head-on collision with agent 3.

Thus, to calculate  $h_3$ , we first calculate  $v_{\text{boost}}$ , the maximum speed achieved by agent 2 after it comes in contact with agent 1.  $v_{\text{boost}}$  can be found by solving

$$\begin{aligned}
v_{\text{boost}} &= \max_{\angle \dot{\mathbf{r}}_2(0) \in [0, 2\pi), t \geq 0} \|\mathbf{r}_2(t)\| \\
\text{subj. to } \mathbf{r}_1(0) &= \mathbf{0}, \dot{\mathbf{r}}_1(0) = (v_0, 0), \\
\mathbf{r}_2(0) &= (\ell_r, 0), \|\dot{\mathbf{r}}_2(0)\| = v_0, \\
\|\mathbf{r}_1(t) - \mathbf{r}_2(t)\| &\leq \ell_r, \text{ Eq. (2.7) holds.}
\end{aligned} \tag{2.12}$$

We find  $\angle \dot{\mathbf{r}}_2(0)$  is around  $90^\circ$  which allows  $\mathbf{r}_1$  to ‘push’  $\mathbf{r}_2$  and increase its speed. After finding  $v_{\text{boost}}$ , the headroom  $h_3$  is similar to  $h_2$ , where the worst case is a head-on collision with  $\|\dot{\mathbf{r}}_2\| = v_{\text{boost}}$  and  $\|\dot{\mathbf{r}}_3\| = v_0$ , thus

$$\begin{aligned}
h_3(\mathbf{p}) &= \min_{t \geq 0} \|\mathbf{r}_3(t) - \mathbf{r}_2(t)\| - 2r \\
\text{subj. to } \mathbf{r}_2(0) &= \mathbf{0}, \dot{\mathbf{r}}_2(0) = (v_{\text{boost}}, 0), \\
\mathbf{r}_3(0) &= (\ell_r, 0), \dot{\mathbf{r}}_3(0) = (-v_0, 0), \\
\text{Eq. (2.7) holds.}
\end{aligned} \tag{2.13}$$

Having defined the headroom  $h_2, h_3$  for 2 and 3 agents respectively, we can now use them to ‘tune’ the collision avoidance and find parameters which are on the edge of safety, that is  $h_2 = 0$  or  $h_3 = 0$ . To do this, we assume that all of the collision avoidance parameters  $\mathbf{p}$  are given except one and solve  $h_n(\mathbf{p}) = 0$  as a numerical root-finding problem, assuming  $n = 2, 3$ . Based on our observations from Figs. 3.2 and 3.3 we believe this will give us the optimum point between emergence and safety. While our headroom approach  $h_n$  works empirically when considering just  $n = 2, 3$ , the complexity of this approach for  $n > 3$  motivates the need to co-design a highly specialized collision avoidance algorithm for this

behavior instead of tuning a generic algorithm. Additionally, guaranteeing safety is difficult due to our simplifying assumptions made in formulating Eq. (2.7). We leave guarantees of safety as well as consideration of  $h_n$  for  $n > 3$  to future work.



## Chapter 3: Results

To validate our theory, we explore many different combinations of the parameters and the choice of collision avoidance to see how the convergence quality  $\lambda$  is affected. For each particular choice of parameters we initialize all agents on a grid formation with a spacing of  $\max(0.1, \ell_r, 5r)$ , set their initial speeds to  $v_0$ , and set their headings randomly. As mentioned in Section 2.2, we *respawn* agents that collide in order to ensure that collisions are actually catastrophic and that the number of agents  $N$  remains fixed.

### 3.1 Comparing Collision Avoidance Algorithms

Through our convergence metric  $\lambda$ , the potential-fields method [10] and the gyroscopic method [11] are compared. To keep the comparison unbiased, we allow each collision avoidance method to take on a range of repulsion strength  $c_r$  between 0 and 4. We then select the value of  $c_r$  that gives the best convergence quality  $\lambda$ . Fig. 3.1 shows  $\lambda$  and the crash rate for both collision avoidance methods as a function of the number of agents, where the other parameters are fixed at  $\alpha = 0.0005, t_d = 2.5, \beta = 1, v_0 = 0.12, \ell_r = 0.6, a_{\max} = 0.6, r = 0.1$ . It is clear that the potential fields strategy is better for this set of parameters.

### 3.2 Choosing Collision Avoidance Parameters

We show in Section 2.3 how to choose parameters for the potential fields strategy which are on the ‘edge of safety’, that is just barely strong enough to avoid collisions. Fig. 3.2 shows a plot of the quality  $\lambda$  and crash rate as a function of two parameters: the agent size  $r$  and force multiplier  $c_r$ , with the other parameters held fixed at  $N = 20, \alpha = 0.001, t_d = 2.5, \beta = 1, \ell_r = 0.5, a_{\max} = 0.6, v_0 = 0.12$ . Additionally, Fig. 3.2 shows two curves defined by the

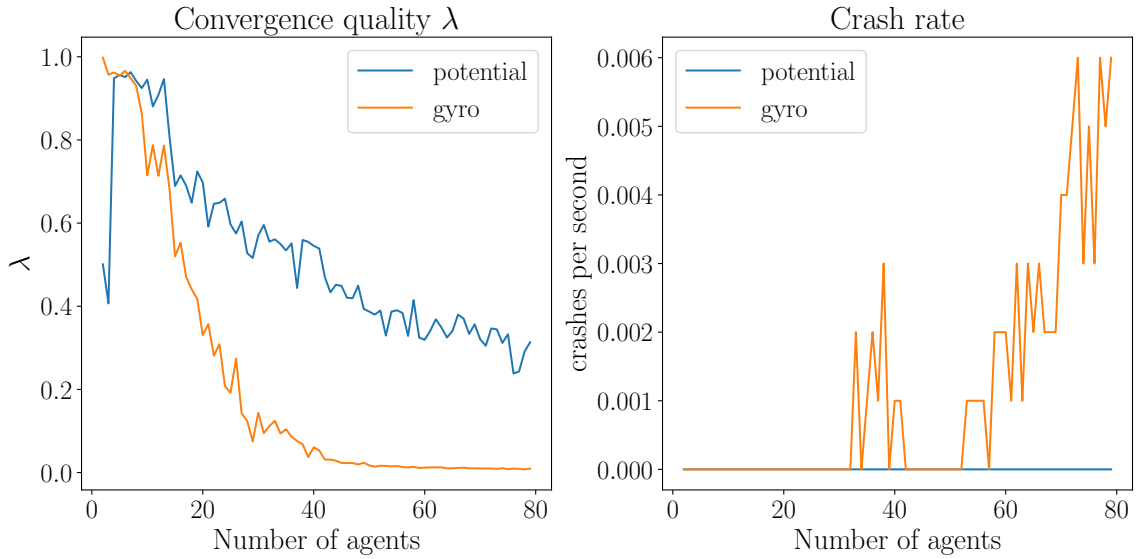


Figure 3.1: A comparison how each collision avoidance strategy scales with the number of agents  $N$ . Left: the convergence quality  $\lambda$ . Right: the crash rate.

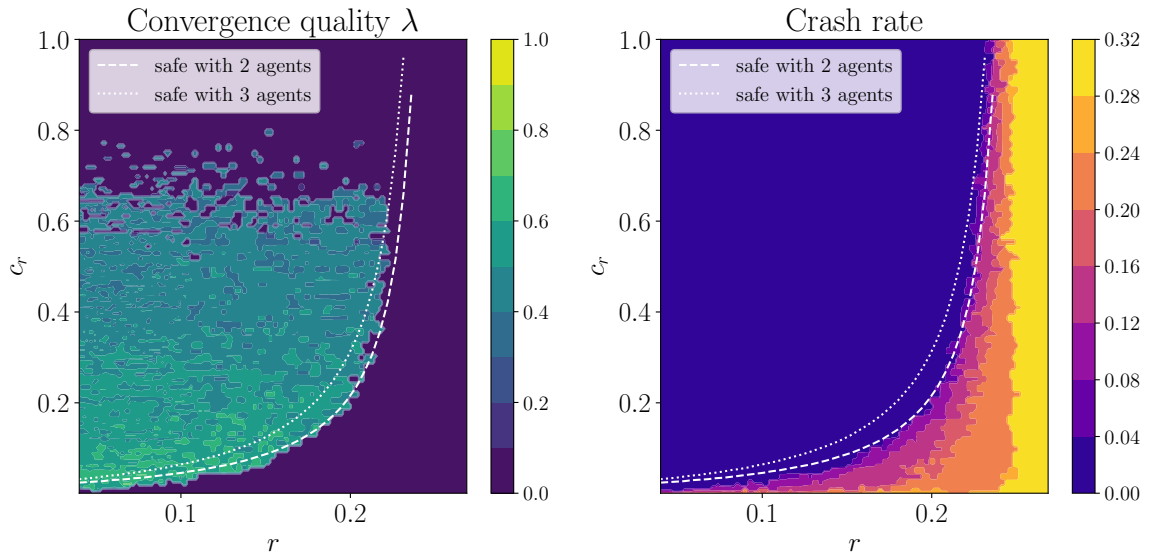


Figure 3.2: Left: The convergence quality  $\lambda$ . Right: the crash rate as a function of  $r$  and  $c_r$ . The white lines show safe values of  $c_r$  as a function of  $r$ , where safety is defined by  $h_2(\mathbf{p}) = 0$  and  $h_3(\mathbf{p}) = 0$ .

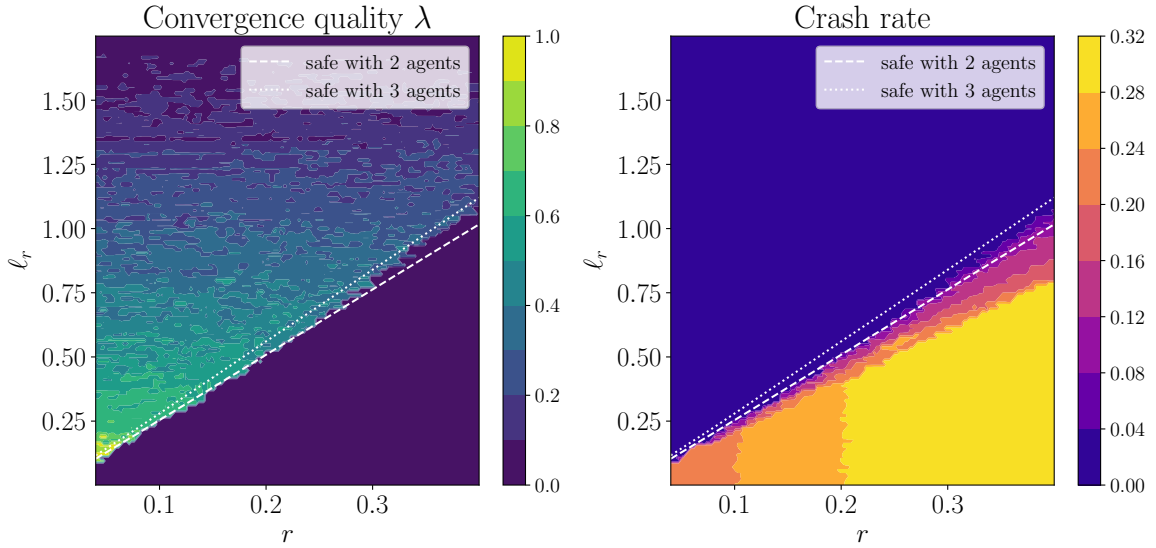


Figure 3.3: A similar plot to Fig. 3.2 except  $c_r = 0.2$  and  $\ell_r$  is varied.

value of  $c_r$  as a function of  $r$  where the headroom  $h_2 = 0$  and  $h_3 = 0$ . Our procedure is able to predict the boundary between  $\lambda = 0$  and  $\lambda > 0$  quite well, with  $h_3 = 0$  giving more conservative parameters that are almost entirely crash-free except for extreme values of  $r$ . Fig. 3.3 shows similar results if we predict the boundary value of  $\ell_r$  instead of  $c_r$ .

It is clear from our results that there is a strong inter-dependency between the choice of collision avoidance and the emergent behavior. Parameters which are ‘below’ the boundary line approximately defined by  $h_2 = 0$  seem to produce no useful behavior due to too many collisions, as can be seen by the crash rate plot on the right side of Figs. 3.2 and 3.3. Parameters which are just ‘above’ the boundary line seem to produce the best results, i.e., the best quality  $\lambda$ , but as we increase the aggressiveness of the avoidance the quality gradually drops away to  $\lambda = 0$  and there is no meaningful emergent behavior as well.

### 3.3 Conclusion

Although research on swarming algorithms and collision avoidance have received significant amounts of attention independently from one another, this paper shows why further research

is necessary in applications where collisions cannot occur. We support our claim through an illustrative example of a particular behavior that is disrupted by different collision avoidance strategies unless great care is taken to tune the collision avoidance parameters. We extend our results in a followup work [19] where we explore two more sophisticated collision avoidance strategies: ORCA [26] and control barrier certificates [13]. This paper thus identifies the need for novel controllers that are co-designed to account for both collision avoidance and the globally emergent behavior simultaneously.

We also show that there is a methodical technique in choosing design parameters which maximize convergence quality yet also avoid collisions and we demonstrate its efficacy empirically. We propose that the best parameters are those which are on the *edge of safety*, or intuitively as weak as possible while being strong enough to avoid collisions.

For the future, we intend to develop novel controllers that can achieve the behavior mentioned here while simultaneously ensuring collision avoidance. We intend to have agents sense each other locally as opposed to the infinite range communication assumption described earlier. Finally, we plan to test additional swarm behaviors to explore how our results generalize, including to three dimensional cases.

## Bibliography

- [1] C. W. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” in *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, New York, NY, 1987, pp. 25–34. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=37402.37406>
- [2] L. Bayindir, “A review of swarm robotics tasks,” *Neurocomputing*, vol. 172, pp. 292–321, 2016.
- [3] E. Bonabeau, D. d. R. D. F. Marco, M. Dorigo, G. Theraulaz *et al.*, *Swarm intelligence: from natural to artificial systems*. Oxford university press, 1999, no. 1.
- [4] J. Carrillo, M. D’Orsogna, and V. Panferov, “Double milling in self-propelled swarms from kinetic theory,” *Kinetic and Related Models*, vol. 2, no. 2, pp. 363–378, 2009.
- [5] Q. Chen, G. Luo, Y. Tong, X. Jin, and Z. Deng, “Shape-constrained flying insects animation,” *Computer Animation and Virtual Worlds*, vol. 30, no. 3-4, pp. 1–11, 2019.
- [6] M. Dorigo and T. Stützle, “Ant colony optimization: Overview and recent advances,” in *International Series in Operations Research and Management Science*, 2019, vol. 272, pp. 311–351.
- [7] H. Wang and M. Rubenstein, “Shape Formation in Homogeneous Swarms Using Local Task Swapping,” *IEEE Transactions on Robotics*, pp. 1–16, 2020.
- [8] Y. Mulgaonkar, A. Makineni, L. Guerrero-Bonilla, and V. Kumar, “Robust Aerial Robot Swarms Without Collision Avoidance,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 596–603, 2018. [Online]. Available: <http://ieeexplore.ieee.org/document/8115305/>
- [9] T. Chung, “Offensive swarm-enabled tactics (offset),” <https://www.darpa.mil/work-with-us/offensive-swarm-enabled-tactics>, 2017, accessed: 2019-09-10.

- [10] K. Szwaykowska, I. B. Schwartz, L. Mier-Y-Teran Romero, C. R. Heckman, D. Mox, and M. A. Hsieh, “Collective motion patterns of swarms with delay coupling: Theory and experiment,” *Physical Review E*, vol. 93, no. 3, p. 032307, 2016.
- [11] Dong Eui Chang, S. Shadden, J. Marsden, and R. Olfati-Saber, “Collision avoidance for multiple agent systems,” in *IEEE Conference on Decision and Control*, vol. 42, Maui, HI, 2003, pp. 539–543. [Online]. Available: <http://ieeexplore.ieee.org/document/1272619/>
- [12] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Springer Tracts in Advanced Robotics*, vol. 70, no. STAR, 2011, pp. 3–19.
- [13] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, “Control Barrier Certificates for Safe Swarm Behavior,” *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68–73, 2015.
- [14] S. H. Arul, A. J. Sathyamoorthy, S. Patel, M. Otte, H. Xu, M. C. Lin, and D. Manocha, “LSwarm: Efficient Collision Avoidance for Large Swarms With Coverage Constraints in Complex Urban Scenes,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3940–3947, 2019.
- [15] A. Breitenmoser and A. Martinoli, “On combining multi-robot coverage and reciprocal collision avoidance,” *Springer Tracts in Advanced Robotics*, vol. 112, pp. 49–64, 2016.
- [16] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, “Optimized flocking of autonomous drones in confined environments,” *Science Robotics*, vol. 3, no. 20, p. eaat3536, 2018.
- [17] L. Dai, Q. Cao, Y. Xia, and Y. Gao, “Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance,” *Journal of the Franklin Institute*, vol. 354, no. 4, pp. 2068–2085, 2017.
- [18] M. R. D’Orsogna, Y. L. Chuang, A. L. Bertozzi, and L. S. Chayes, “Self-propelled particles with soft-core interactions: Patterns, stability, and collapse,” *Physical Review Letters*, vol. 96, no. 10, 2006.
- [19] C. Taylor, C. Luzzi, and C. Nowzari, “On the effects of collision avoidance on an emergent swarm behavior,” *arXiv preprint arXiv:1910.06412*, 2019.
- [20] Y. li Chuang, M. R. D’Orsogna, D. Marthaler, A. L. Bertozzi, and L. S. Chayes, “State transitions and the continuum limit for a 2D interacting, self-propelled particle system,” *Physica D: Nonlinear Phenomena*, vol. 232, no. 1, pp. 33–47, 2007.
- [21] S. V. Viscido, J. K. Parrish, and D. Grünbaum, “The effect of population size and number of influential neighbors on the emergent properties of fish schools,” *Ecological Modelling*, vol. 183, no. 2-3, pp. 347–363, 2005.
- [22] A. Cavagna, A. Cimarelli, I. Giardina, G. Parisi, R. Santagati, F. Stefanini, and M. Viale, “Scale-free correlations in starling flocks,” *Proceedings of the National Academy of Sciences*, vol. 107, no. 26, pp. 11 865–11 870, 2010. [Online]. Available: <http://www.pnas.org/cgi/doi/10.1073/pnas.1005766107>

- [23] Y. Xiang, D. Sun, W. Fan, and X. Gong, “Generalized simulated annealing algorithm and its application to the thomson model,” *Physics Letters A*, vol. 233, no. 3, pp. 216–220, 1997.
- [24] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” *Journal of global optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [25] E. Jones, T. Oliphant, P. Peterson *et al.*, “SciPy: Open source scientific tools for Python,” 2001–, [Online; accessed 2020-03-13]. [Online]. Available: <http://www.scipy.org/>
- [26] J. Van Den Berg, J. Snape, S. J. Guy, and D. Manocha, “Reciprocal collision avoidance with acceleration-velocity obstacles,” *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3475–3482, 2011.