Learning Collective Behavior in an Experimental System of Feedback-Controlled Microswimmers

Doctoral thesis for obtaining the academic degree of Doctor of Natural Sciences

submitted by

Robert Clemens Löffler

to the

Universität Konstanz
Mathematisch-Naturwissenschaftliche Sektion
Fachbereich Physik

Konstanz, 2023
Acknowledgments

This work would not have been possible without the help of many people. First and foremost, I would like to express my gratitude towards my supervisor Professor Clemens Bechinger for accepting me in his group and enabling me to carry out this project. I am deeply thankful for his continuous support and valuable advice and guidance throughout all stages of my research and dissertation. I also would like to greatly thank my predecessor Tobias Bäuerle, who designed and build the initial version of the experimental setup used throughout this work and from whom I have learned a lot during the time of our collaboration in the first two years of my research. Equally, I would like to express my special thanks to Emanuele Panizon, who introduced me to the field of reinforcement learning, and contributed to the experiments presented in Chapters 7 and 8. He not only supported me with many fruitful scientific discussions and his always calm and reassuring advice, but also wrote a substantial part of the program code which is implementing the framework of reinforcement learning in our experiment control software. Likewise, I am grateful to our collaborators Christian Rohwer and Professor Mehran Kardar, who derived the theoretical model shown in Chapter 6, which significantly improved our understanding of the system and our observations. I would also like acknowledge Professor Matthias Fuchs for reviewing this thesis and agreeing to chair my examination committee. Furthermore, I would like to thank Chun-Jen, Veit and all other members of the Bechinger group for many lively discussions, good advice and overall support. I would similarly like to extend my thanks to all other persons in the physics department who helped me or otherwise supported me at some point during my thesis. Finally, I am extremely grateful to my family and friends, but most of all Zoë for continuously encouraging and supporting me whenever necessary. Thank you, everyone!
Abstract

Collective behavior in groups is a recurring phenomena in nature. It is present on vastly different length and time scales and can occur in small groups of few individuals as well as in colonies of millions of participants. Over the past few decades, research effort has been increased to better understand and also model the wide variety of observed collective patterns. While modern computer simulations have helped the effort, at the same time experimental model systems of artificial active matter have become a compelling middle ground. They can enable precise control over interactions between agents while providing the complexity of real environments, thus, bridging the gap between nature and simulations. Similarly, different concepts of modeling the decision process of collectively acting agents have been developed. A common approach is to apply simple interaction rules based on so-called social forces, which produce rich collective behavior and can also lead to criticality in the observed dynamics. More recently, though, an effort has also been started to model the evolutionary process leading to such rules, by employing techniques from multi-agent reinforcement learning.

In the experiments presented in this thesis, we apply both social interaction rules as well as reinforcement learning to an experimental model system of artificial light-activated microswimmers, for which we are able to individually manipulate the speed and orientation by an external feedback control loop. First, a social interaction rule is applied to a group of active particles, which can lead to unordered swarming and rotationally ordered swirls, tuneable by a single angular parameter. Continuously varying this parameter, we find a continuous transition between swarms and swirls, with a clear bifurcation point in rotational symmetry, which is spontaneously broken into either direction above the critical angle. Furthermore, we provide a minimal model for this behavior, which describes the observations well by simple symmetry arguments without the assumption of thermal equilibrium. We further verify the continuous nature of the transition by experimental measurements of a distinct hysteresis loop. At the same time, we are also able to recreate swirling motion by reinforcement learning, based on a reward function inspired by social forces, to discuss similarities and differences between the two approaches. Notably in the case of reinforcement learning, the symmetry in rotational order is not broken spontaneously during motion, but in an early state of the learning process, making the broken symmetry a crucial part of the behavioral
policy. Finally, we discuss a more general reinforcement learning scenario, where the reward for individuals is solely provided by feeding on a virtual food source. Despite the selfish nature of the task, collective behavior emerges in the group, including swirling motion as seen in the previous experiment. We find that this collectivity is mainly driven by the complex interactions between agents, regarding information transfer as well as physical interactions between microswimmers in the experimental system and can only partially be replicated by auxiliary simulations. The resulting policy is robust enough to even provide stable collective motion when applied to an previously unseen scenario in which food is completely absent.

Our results highlight the importance of model experiments for social interaction rules as well as in reinforcement learning. Experiments come with a complexity which is hard to replicate in simulations but might be vital for the emergence of robust mechanisms in collective behavior. Considering and understanding these solutions is not only beneficial to better understand trade-offs in natural systems, but is equally important when designing future artificial systems of autonomously acting agents, where the existence of a “reality gap” is well known and much discussed in literature.
Zusammenfassung


Unsere Ergebnisse machen deutlich wie wichtig experimentelle Modellsysteme sind, sowohl für das Verständnis von sozialen Interaktionsregeln als auch bei der Anwendung maschinellen Lernens. Experimente besitzen eine natürliche Komplexität, die sich in Simulationen nur schwer nachbilden lässt, die aber im gezeigten System durchaus entscheidend für die Entwicklung verschiedener robuster Mechanismen in kollektivem Verhalten ist. Diese Komplexität zu berücksichtigen und zu verstehen ist dabei nicht nur unabdingbar um Verhaltensabwägungen in biologischen Systemen besser nachvollziehen zu können, sondern gleichermaßen wichtig beim Entwurf künstlicher Systeme aus autonomen Einheiten, ein Feld auf dem die sogenannte „Realitätslücke“ hinreichend bekannt ist und oft diskutiert wird.
# Contents

**Acknowledgments**  
3

**Abstract**  
5

**Zusammenfassung**  
7

1 **Introduction**  
11

2 **Principles of Collective Behavior**  
15
   2.1 Classical models of collective motion  
   16
   2.2 Collective behavior as a result of optimization  
   20

3 **The Concept of Multi-Agent Reinforcement Learning**  
21
   3.1 Markov decision processes  
   22
   3.2 Policy approximation  
   24
   3.3 Application to multi agent systems  
   27

4 **Active Matter as a Model System**  
31
   4.1 Artificial microswimmers  
   32
   4.2 Active Brownian motion  
   33
   4.3 Dense active matter  
   34
   4.4 Strategies of external control  
   35

5 **Fabrication and Control of Synthetic Microswimmers**  
37
   5.1 Preparation of active particles  
   39
   5.2 Feedback control for individual particle steering  
   39
   5.3 Calibration of laser illumination  
   43
   5.4 Implementation details of propulsion and rotation  
   44

6 **Swirls of Active Particles based on Social Interaction Rules**  
47
   6.1 Interaction rule  
   48
Introduction

Collective behavior in groups of animals and insects is a recurring phenomenon in nature. It is present on vastly different spatial and temporal scales and can range from groups of few individuals to colonies of millions of participants. Typical examples are schooling of fish (Parrish et al. 2002), mass migration of locust (Buhl et al. 2006), flocking of birds (Cavagna et al. 2010) and swarming of midges (Cavagna et al. 2017). The richness and diversity of the manifestation of such collective groups is the main motivation behind the field of collective behavior. The common questions which are tackled in this field of research pertain both the “why” of such collectivity, e.g. possibility of protection from predation, benefits in foraging as well as efficient migration and the “how”, e.g. how information flows in flocks (Hemelrijk and Hildenbrandt 2011) or how swirls can be achieved in fish banks (Couzin et al. 2002).

The question of “how” has been a focus of research for the past few decades. A lot of work has been dedicated to reproduce collective dynamics using simple interaction rules based on local so-called social forces (which, on the contrary to physical forces, can be non-reciprocal and non-monotonic) that lead to collective behavior of the group similar to those of living systems. These forces include attraction to, alignment with, as well as repulsion from other individuals or other features of the surroundings and are based on the local environment, like the position and orientation of neighboring agents or varying sorts of sensory inputs (Vicsek et al. 1995; Couzin et al. 2002; Barberis and Peruani 2016; Delcourt et al. 2016). Such interactions can effectively also transmit local information from agents to neighboring agents and substitute communication (Hemelrijk 1990; Hemelrijk and Hildenbrandt 2011; Attanasi et al. 2014a). Applications of such models have been fundamental to describe behavior in-situ as well as in simulations of model-systems which offer a more precise control of the environment and possible interaction of individuals (Tunstrøm et al. 2013; Feinerman et al. 2018).

Parallelly, the question of “why” has also been actively researched. However, while it is possible (despite often quite difficult) to analyze motion of living systems it is generally challenging
1 Introduction
to quantify intrinsic motivation in a meaningful way. This is where the concept of multi-agent reinforcement learning (MARL) has brought a novel approach, since it allows to study behavioral strategies as the optimal response to a task (Hahn et al. 2019; Sunehag et al. 2019; Durve et al. 2020; Gerhard et al. 2021). Instead of studying the behavior of motion as created by a priori rules, with MARL it is the task that determines the final motion as the result of an unbiased optimization process.

In this work, we apply both social interaction rules as well as a reinforcement learning framework to an experimental model system of light-activated artificial microswimmers. Microswimmers have proven to be an excellent model system to study emergent collective phenomena, where some of the interaction rules can be encoded in physical properties. Systems like Quinke rollers (Bricard et al. 2013; 2015) and ferromagnetic rollers (Kaiser et al. 2017; Kokot and Snezhko 2018) have been studied for their ability to align, forming flocks and swirls depending on the confinement. Light-activated active particles on the other hand can be used to switch motility by external control depending on interaction rules (Bäuerle et al. 2018; Lavergne et al. 2019). Similarly, reinforcement learning has been studied in the context of microswimmers before on single particles (Muiños-Landin et al. 2021) and in simulations (Gerhard et al. 2021), however, sufficient experimental control to apply reinforcement learning to multiple particles has not been achieved, yet. Here, I present a novel approach to not only control the speed of light-activated synthetic microswimmers, but also to individually induce active torque on these particles via a feedback-loop system. We use this fine-grained level of control to apply both classic models of collective behavior based on social forces as well as experimental multi-agent reinforcement learning to a set of 30-50 active particles. Applying MARL to microswimmers is not only of interest to compare different approaches in modeling collective behavior, but also for the wider field of microswimmer applications and micro-robotics, as we present the first instance where training is performed with multiple agents in such a microscopic experimental system.

Outline. Following this introduction, Chapter 2 provides a brief overview of the field of collective behavior. It discusses relevant models based on interaction rules and their implications, namely the presence of dynamic group phases and the transitions between these, but also motivates the use of MARL as an alternative approach. Chapter 3 provides the reader with all necessary theory to understand the framework of reinforcement learning and to follow the steps of the optimization algorithm, focusing on the recent advancements which are applied in the context of this work. Next, Chapter 4 introduces the field of soft active matter and more specifically the recent development in the area of artificial microswimmers and their use as a model system to study collective behavior. Our own implementation of such a model system is presented in Chapter 5, which describes the
preparation of synthetic microswimmers, our experimental setup, the feedback control loop, its interaction with the reinforcement learning algorithm and our methods for fine-grained steering of active particles.

Chapter 6 covers our observations and results from applying a constant interaction rule based on social forces to a group of microswimmers. Depending on a single parameter, the rule is able to create an unordered swarming phase as well as a rotationally ordered swirling group. Tuning this parameter, we observe symmetry breaking in particle alignment which we are able to describe by a simple dynamical model, identifying a critical bifurcation point. Furthermore, when biasing the preferred direction of rotation of the swirling group, we can experimentally measure hysteresis above the bifurcation point. In order to compare this interaction rule to an approach based on reinforcement learning, Chapter 7 investigates an task designed to lead to comparable swirling motion. While achieving similar behavior of a rotating particle group, the most notable difference is that for multiple reasons the symmetry breaking does not happen spontaneous in the final motion but occurs during the training process. While the swirl in Chapter 7 is created by a rather artificial task, representing some kind of “social potential”, the task to be achieved by the active particles in Chapter 8 is more simplistic and inspired by natural systems. A (virtual) food source is added to the experiment and individual agents are rewarded for collecting this food. Despite the selfish nature of the task, we observe emergent collective behavior, including again swirling motion, due to the complex physical environment of the microswimmers. Finally, Chapter 9 concludes on the presented results and discusses the more general implications of our findings.

**Overlapping publications.** Most results presented in this thesis have already been or will eventually be published in independent research articles:

“Formation of stable and responsive collective states in suspensions of active colloids”.
*Nature Communications* 11, p. 2547.

“Behavior-dependent critical dynamics in collective states of active particles”.

Löffler, R. C., E. Panizon, and C. Bechinger (2023).
“Collective foraging of active particles trained by reinforcement learning”.
Manuscript submitted.
Principles of Collective Behavior

Collective behavior as a general concept describes the often spontaneous emergence of larger scale processes or motions within a group of individuals. While the term “collective behavior” was originally used in human sociology (Smelser 1962) the concept has since been adopted to describe living, and more recently also artificial systems of interacting individuals in general. In fact, collective behavior in nature can be observed on all spatial and temporal length scales. Well known examples range from bacteria (Czirók et al. 1996) to fish (Parrish et al. 2002), birds (Cavagna et al. 2010) and mammals (Ginelli et al. 2015) and can include few individuals or colonies of millions (Buhl et al. 2006). Similarly, relevant time scales for collective behavior range from a few seconds, e.g. in escape motions, to full life cycles in collective evolutionary processes (Ioannou and Laskowski 2023). All those examples have in common, that the observed behavior manifests in patterns of collectivity which can be coarse-grained and described on the group scale.

While general motivations for cooperative behavior like better protection from predation, mating or increased foraging efficiency (the why) are well known in biology (Sumpter 2010), it is often difficult to link a specific motivation to observed patterns of collective behavior. Similarly, the question often remains exactly how the interactions between individual group members with each other work to achieve the observed patterns, what amount of local information is necessary for an individual to participate in collective behavior and how universal such interactions can eventually be (Sumpter et al. 2008; Vicsek and Zafeiris 2012).

In this chapter, I will first give an overview of the common approach to modeling collective behavior, which is through simple interaction rules, based on so called social forces. Namely, these include the Vicsek model and various versions of the zonal model. I will further discuss the resulting collective states and the often discussed hypothesis of collective systems operating close to a critical point (Muñoz 2018), with all implications of such a regime. Lastly, a different approach is presented, investigating collective interactions as the result of an optimization process in order to better link motivation to the observed behavior.
2 Principles of Collective Behavior

2.1 Classical models of collective motion

Most models of collective behavior are based on fixed interaction rules: An agent, a single individual in a larger group, is subject to *social forces* which determine its motion within the group. Common social forces are attraction to or repulsion from other group members as well as the tendency to align with the orientation of other group members. Importantly and contrary to physical forces, these social forces do not need to be reciprocal (Hemelrijk 1990). For example, in a vision based model, one agent might “see” another agent and act accordingly, but not vice-versa. Also, more complex models might include different sub-species within one group which do not mutually interact with each other. The classic example here is a predator-prey situation, in which the predator is attracted to the prey, while the prey on the other hand is well advised to be repelled by the predator. Depending on the forces acting on an individual agent, it is adjusting its speed, direction of motion, or what other controls are available in the given system.

A first approach on modeling collective behavior in such a way has been presented by Reynolds (1987) in his work on modeling bird flocks. On top of a rather detailed physical model of bird flight in 3D space, he introduced three behavioral interaction rules: Based on perception, each agent tries, first, to avoid collisions with nearby neighbors, second, attempts to match its velocity to close neighbors and, third, tries to stay close by to the general flock.

Vicsek model. A more generalized approach to collective motion is the well known Vicsek model, where agents in a periodic 2D plane solely interact by aligning themselves to close neighbors selected by a fixed interaction radius (Vicsek et al. 1995). Single agents are considered to be point like particles with a constant velocity $v_0$ in the direction of orientation $\theta_i,t$. The system is evolved in an Euler-scheme with time step $\Delta t$ where the orientation of each particle in the next step is based on the average orientation of neighbors $j$ within a given radius $R$ and additional linear noise $\delta \theta \sim [-\eta/2, \eta/2]$. Thus,

$$x_{i,t+1} = x_{i,t} + v(\theta_{i,t}) \Delta t, \tag{2.1}$$

$$\theta_{i,t+1} = \langle \theta_{j,t} \rangle_{|x_i - x_j| \leq R} + \delta \theta_i. \tag{2.2}$$

Interestingly, this simple general model comprises a phase transition. While for low density and high noise, particles move randomly within the plane, for high densities and low noise a long-range orientational order is developed. A theoretical description for this phase transition has been found by Toner and Tu, who introduced the interpretation as a dynamic $XY$-model for which they
2.1 Classical models of collective motion

Figure 2.1: Zonal Model. An agent with a distinct orientation (denoted by the arrow head) is guided by neighboring agents in three zones: A zone of repulsion (zor) to avoid collisions between agents, a zone of orientation (zoo) which is used to align with other group members and a zone of attraction (zoa), keeping the group cohesive on a larger scale. The group behavior heavily depends on the size and weights of the zones in relation to each other. Additionally zones do not need to be uniformly around the agent, as an example, this sketch includes a blind angle to the back of the agent.

were able to derive a continuum description exerting the unordered to long-range orientationally-ordered phase transition (Toner and Tu 1995; Toner et al. 2005). Notably, this phase transition exists in the absence of thermodynamic equilibrium, as the individual particles have by definition of the model an intrinsic velocity and, thus, perform constant work on their surrounding environment.

Zonal models. While the Vicsek model is well-known in physics for its novel phase transition, it fails to model most of the collective dynamics which are observed in natural systems. Reducing this discrepancy is a target of zonal models, first introduced by Couzin et al. (2002) to model schooling fish, where every agent reacts to multiple social forces exerted by neighboring group members in different zones. Commonly, these zones include a small zone of strong repulsion to enforce collision avoidance, a zone of alignment for motion coordination and a zone of attraction to retain cohesion of a larger group as a whole (Fig. 2.1). As these models are based on a concept of vision of individual agents, zones can also include blind angles in which neighboring particles have no effect as well as a finite vision horizon. While the repulsive force of the inner zone often takes precedence over any other interaction, alignment and attractive forces are usually but not always weighted equally. Depending on the exact ratio of these weights as well as the spatial extensions of the different zones, several phases of cohesive group behavior can emerge: An unordered swarming phase, mainly governed by attraction of individuals, which is cohesive but does not show further correlation between agents, a polarized flocking phase, primarily governed by alignment, in which all individuals are heading in the same direction, and a swirling or milling phase, where agents are aligned tangentially, resulting in a high degree of rotation of the group (Couzin et al. 2002; Gautrais et al. 2008). Moreover the model is also able to reproduce transitions between these states and compares well with experimental observations in actual schools of fish, depicted in Fig. 2.2 (Tunstrøm et al. 2013).
Collective states. Note, that the three states of swarming, flocking and swirling/milling discussed above are not specific to fish, but can be seen a basic states of collective motion in general. While all three of them describe cohesive groups of individuals, they differ in their degree of local and global order. A swarm describes a cohesive but fully unordered group of individuals with “chaotic” internal structure. At the same time, this also represents a stationary group state, where the global speed of the group is negligible. Situated on the other side of the spectrum is the polarized flock, with a fully broken orientational symmetry. That is, all agents are aligned with a global direction and the group is moving at maximum speed, akin to the ordered states found in the models of Vicsek et al. (1995) and Toner and Tu (1995). In a swirl, on the other hand, there is no global directional order and the group is stationary, however, agents do have local order, i.e. they are highly aligned with their direct neighbors. The latter is an important property especially for dense natural systems of swimming or flying individuals where it is necessary to avoid collisions between group members. Therefore it is of advantage to keep a high degree of local alignment, independent of the fact if the group as a whole would like to keep its position or move somewhere else. Even though the zonal model discussed above assumes point-like particles, the importance of this collision avoidance is represented by the precedence of repulsion from other agents over any other social forces.

Proximity to critical transitions. While ordered states like swirls and flocks are commonly observed in natural systems, a often debated hypothesis suggests that most of these systems might operate close to the edge of stability or, in other words, close to a critical point of a phase tran-
2.1 Classical models of collective motion

Transition into other collective states (Bak 1996; Muñoz 2018). Being close to criticality might have several advantages to the dynamics of collective groups and might provide a optimal balance between robustness and flexibility to adapt to changing situations and environments. From a physical perspective, being close to a critical transition will increase general susceptibility and relaxation times, leading to more pronounced responses to external stimuli to the group, e.g. a predator. At the same time, the increase in spatial correlation lengths should also result in faster transfer of information traveling through the group, like a change of direction in a flock of birds. In fact, scale free behavior suggesting proximity to criticality has for example been reported for large flocks of starlings (Cavagna et al. 2010) as well as wild swarms of midges (Attanasi et al. 2014b; Cavagna et al. 2017). While criticality would imply a continuous phase transitions, other aspects of natural systems can also be described by multi-stability of different collective states in what can physically be described as a “coexistence regions” (Calovi et al. 2014). Nevertheless, one should be reminded that biological systems are highly dynamic and often of limited size, so the applicability of concepts from equilibrium statistical mechanics should be considered carefully.

A different aspect of collective behavior which is not represented in the models discussed so far, is a dispersity of information of individual agents and how collective decisions are formed within the group. A common theme is that social interaction rules facilitate consensual decision making, for example on the direction the group is traveling, based on symmetry breaking in the collective motion (Conradt and Roper 2005; Ward et al. 2008). Taking external stimuli into account, however, different agents in the group might have different priorities underlying their social forces. If only a part of the group is aware of for example a nearby predator, these individuals might start an escape motion, taking over effective leadership of the group (Couzin et al. 2005). Having the aforementioned large correlation lengths might be of immediate advantage here, as a minority of agents with a strong bias can sufficiently influence the motion of the group as a whole, without the necessity to spread the actual information by means of direct communication.

While the arguments given above conceptually assume equality of all group members, this is neither necessarily the case in biological systems, where natural variance in individuals and complex social dynamics are at play. As a result of a hierarchical structure, some individuals might be more influential to their mates than the other way round, effectively leading to another type of non-reciprocity in social forces. The subsequent bias to collective behavior, has for example been observed in flight patterns of a pigeon flocks (Nagy et al. 2010). Local variability in movement behavior has also been shown to be a major driver for the formation of complex shapes and patterns in flocks of starlings (Hemelrijk and Hildenbrandt 2011), underlining the strong influence of natural variance of individuals to collective behavior in general.
2 Principles of Collective Behavior

2.2 Collective behavior as a result of optimization

While collective behavior can be modeled by simple interaction rules, ultimately these rules result from evolutionary adaptation or cognitive learning processes. Modeling this optimization process as a whole, can lead to helpful insights to better link specific motivation to behavioral patterns. Depending on the complexity of the system and the objective to be optimized, there are different approaches to the problem. A classical example for an optimization objective is the avoidance of direct contact to neighboring individuals, while keeping cohesion of the group e.g. in a flock of birds. In this case, optimal stochastic control theory has been successfully applied to derive an analytic mean field solution for agents to align with each other to achieve the desired behavior (Borra et al. 2021). For less specific tasks, however, where the structure of the interaction rule can be more complex and is not known a priori, purely analytical approaches become less viable.

Here, the framework of multi-agent reinforcement learning (MARL) opens opportunities where complex interaction rules can be optimized through iterative training. Within the framework, the interaction rule with which a given agent reacts to its environment is often encoded by artificial neural networks, which are able to fit complex nonlinear functions of arbitrary dimensions. A lot of focus in prior work employing MARL in context of collective behavior has been laid on studying emergent flocking in predator escape scenarios (Morihiro et al. 2006; Y. Yang et al. 2017; Hahn et al. 2019; Sunehag et al. 2019; Young and La 2020) but also for example to optimize group behavior in different foraging scenarios (López-Incera et al. 2020). All these works have in common, that they mostly represent a evolutionary process of the group, developing the investigated behavior. A slightly different approach has been taken by Durve et al. (2020) who similarly study the development of schooling in fish in order to keep cohesion of the group, but also investigate how a single individual can learn from and adapt to a group of teachers who behave in a predefined manner.
3 The Concept of Multi-Agent Reinforcement Learning

There has been a constant research interest to better understand and model general decision processes and to create “artificial intelligence”. One branch of artificial intelligence which became increasingly important over the last decade is reinforcement learning (RL), which connects concepts of artificial intelligence with techniques of machine learning. Conceptually, reinforcement learning is focused on finding an optimal strategy for an individual agent or player to interact with a given environment in a scheme of actions and responses. Notable achievements over the last years contain general improvements in robotic control (Kober and Peters 2012) as well as mastering the game of Go (Silver et al. 2016; 2017) or multi player card games (Brown and Sandholm 2019). The most recent example is the heavily debated chat bot ChatGPT (OpenAI 2022), for which reinforcement learning based on real human feedback has been used to refine the output of a previously existing language model build by other machine learning techniques (Ouyang et al. 2022; Gao et al. 2022).

Classical reinforcement learning assumes a single agent to act on an environment which responses may be stochastic, but are well defined. However, many scenarios – especially games as mentioned above – have multiple agents, either cooperating or competing, and the resulting system may be more complex, for example non stationary. Research effort to extend the existing frameworks to incorporate multiple agents has therefore equally increased over the last years (Gupta et al. 2017; K. Zhang et al. 2021). This includes research in collective behavior, where multi-agent reinforcement learning (MARL) has been used to investigate the development of flocks (Sunehag et al. 2019; Durve et al. 2020) as well as cooperative strategies for foraging (López-Incera et al. 2020) and predator avoidance (Hahn et al. 2019; Young and La 2020), as already discussed in the previous chapter.

For the sake of simplicity, this chapter first introduces the basics of a single agent system in the context of model-free reinforcement learning with discrete time steps and actions. We then discuss,
what considerations have to be made when expanding this scheme to multiple agents and how the presented techniques are implemented in the context of this thesis.

3.1 Markov decision processes

Broadly speaking, reinforcement learning is an optimization problem where the strategy of an agent is optimized under a certain reward. The problem itself is defined in the form of a Markov decision process: At each time step $t$ the agent is choosing an action $a_t$ solely based on the current states $s_t$ of the environment. As a response to the chosen action $a_t$, the environment may change its configuration into a different state $s_{t+1}$. Furthermore, a reward $r_t$ is associated with each action $a_t$ and the corresponding change in state $s_t \rightarrow s_{t+1}$ (Fig. 3.1). Note, that the response of the environment may be a stochastic process. Consequently, the probabilities for a certain change in state and associated reward are distributed and captured by the so called environment dynamics $p(s', r | a, s)$.

A concrete example of a Markov decision problem is the classic computer game “snake”, which is depicted in Fig. 3.2. A snake, controlled by the player, can move through a grid world, one field at a time. Within the grid world food items are randomly placed in single fields. The goal of the game is to collect as many food items as possible with the snake. However, the game ends, if the snake collides with itself or the boundary of the grid world. In this example, the state $s_t$ of the system captures the full configuration of the grid world, including the exact position of all elements of the snake, as well as the position of the food item. At any step, the player has to choose into which direction the path of the snake is continued (marked with blue arrows in Fig. 3.2). In accordance with the goal of the game, a positive reward is given to the agent for every collected food item. Note, that in the specific case of snake the environment dynamics $p(s', r | a, s)$ is a deterministic function, except for the instance where the current food item is eaten by the snake and a new food item is randomly placed in the environment. The sequence of visited states, chosen actions and

![Figure 3.1](image-url) Interaction between the agent and the environment in a Markov decision process. The agent chooses an action $a_t$ according to its policy $\pi(a | s)$. The environment responds by changing to a new state $s_{t+1}$ and rewarding the agent for its action according to the environment dynamics $p(s', r | a, s)$. Freely adapted from (Sutton and Barto 2018).
Figure 3.2: The computer game “snake”: A playing agent can control the motion of a snake within a finite sized grid world. The goal of the game is to collect as many randomly appearing food items as possible, without colliding the snake with its own tail or any other obstacles.

resulting rewards \( s_0 \rightarrow a_0 \rightarrow s_1, r_0 \rightarrow a_1 \rightarrow \ldots \) within one episode of playing the game is called a trajectory.

In the framework of reinforcement learning, the strategy of an agent playing the game is captured by its so called policy \( \pi(a \mid s) \). While a human player will likely have an intuition that collecting a food item might be positively rewarded, in general there is no a priori knowledge about the environment available to the playing agent. As a result, when starting the learning process, the agent should perform random actions in order to explore the environment and gather experience about it. However, at a certain point the agent has to start to exploit its knowledge gained about the environment in order to increase the collected reward. After all, the task of reinforcement learning is to converge to the optimal policy \( \pi^* \) which maximizes the reward gained by the agent.

The balance between exploration and exploitation is therefore a key aspect of any reinforcement learning algorithm: If more exploration is performed than necessary, the algorithm will take longer to converge towards the optimal policy. However, if the algorithm is to greedy, it might get stuck in an local optimum and will not be able to find the overall best policy.

To formally define the performance of a given policy, we introduce the so called return \( G_t \). It is the cumulative sum over all future rewards of an episode, starting at time \( t \),

\[
G_t = \sum_{t'=t}^{T} v(t'-t) r_{t'},
\]

and serves as the main optimization objective of reinforcement learning, akin to a loss- or cost-function commonly found in other optimization problems. Here, of course, the target is to maximize the return. The discount factor \( \gamma \) is important, as some Markov decision processes can have an infinite episode length \( T \rightarrow \infty \). This is actually the case for the example of snake, assuming that the snake does not grow and the player is not causing any collision. Based on this definition of return we can now define a value for a given state \( v(s) \) and a value for a performed action \( q(a,s) \)
3 The Concept of Multi-Agent Reinforcement Learning

by the expected return (i.e. expected future reward) under the current policy \( \pi \) starting from the given state,

\[
v_\pi(s) = E_\pi[G_t | s_t] = \sum_a \pi(a | s) q_\pi(a, s),
\]

(3.2)

\[
q_\pi(a, s) = E_\pi[G_t | a_t, s_t] = \sum_{s', r} p(s', r | a, s) (r + \gamma v_\pi(s')).
\]

(3.3)

The recursive definitions of \( v_\pi \) and \( q_\pi \) given in Eqs. (3.3) and (3.5) are also known as Bellman equations. Note, that by definition the expected return heavily depends on the current policy. It is mathematically proven for a Markov decision process, that there exist one or more optimal policies \( \pi^* \) that share the same maximal values for all states and actions (Sutton and Barto 2018),

\[
v^*(s) = \max_{\pi} v_\pi(s) \quad \forall s, \quad q^*(a, s) = \max_{\pi} q_\pi(a, s) \quad \forall a, s.
\]

(3.6)

Again, the task of reinforcement learning is to find that optimal policy.

3.2 Policy approximation

There are two fundamentally different approaches on how to find the optimal policy. One method which is often used, is to approximate the optimal action value function \( q^*(a, s) \) through the experience gathered by exploring the environment. If the problem consist of a finite set of discrete \( a \) and \( s \), the implementation of the value function can be as simple as an \( a \times s \) matrix, often called \( q \)-matrix. The second method is to directly approach the optimal policy. Algorithms that use this method are referred to as policy approximation or policy gradient methods. For problems with a continuous state or action space, the policy needs to take some functional form \( \pi(a | s, \theta) \) with parameter set \( \theta \). Here, a neuronal network is often chosen, as it can fit nonlinear functions with arbitrary input and output dimensions without prior knowledge about their exact form. Depending on the amount of intermediate layers, this is often referenced as deep learning. The algorithm then tries to fit the optimal policy using the gradient of some performance measure \( J(\theta) \) to improve the policy parameters,

\[
\theta_{t+1} = \theta_t + \alpha \nabla J(\theta),
\]

(3.7)
where $\alpha > 0$ is the step size. For an episodic case, the necessary performance measurement can be defined as the value / expected return of the initial state of each episode

$$J(\theta) = v_\pi(s_0) = E_\pi[G_0].$$

(3.8)

In order to optimize the policy, we need to derive the gradient of this stochastic property. The policy gradient theorem states that

$$\nabla J(\theta) = \nabla v_\pi(s_0) \propto \sum_s \mu(s) \sum_a q_\pi(a, s) \nabla \pi(a | s, \theta),$$

(3.9)

where $\mu(s)$ denotes the on-policy distribution under $\pi$, i.e. the probability distribution of how often each state is visited within an episode under the current policy. Given this form, the gradient of the performance measure now only depends on the gradient of the policy itself, which we have defined in a functional form in the first place (Sutton and Barto 2018).

**Stochastic gradient ascent.** As the sum over a probability distribution is equal to the expected value of sampling this distribution, we can restate the policy gradient theorem as

$$\nabla J(\theta) \propto \sum_s \mu(s) \sum_a \nabla \pi(a | s, \theta) \left[ \frac{q_\pi(a, s)}{\pi(a | s, \theta)} \nabla \pi(a | s, \theta) \right]$$

$$= E_\pi \left[ q_\pi(s_t, a_t) \nabla \log \pi(a_t | s_t, \theta) \right]$$

$$= E_\pi \left[ G_t \nabla \log \pi(a_t | s_t, \theta) \right].$$

(3.10)

The form of the last equation gives rise to the stochastic gradient ascent step

$$\theta_{t+1} = \theta_t + \alpha G_t \nabla \log \pi(a_t | s_t, \theta)$$

(3.11)

of the classical “REINFORCE” algorithm (Williams 1992). Note that this algorithm relies on the actual return $G_t$ of each action, so the updates can only be applied retrospectively at the end of each episode. However, the update can be intuitively understood, as the gradient in Eq. (3.11) points to the direction in the parameter space that increases the probability for action $a_t$ which is weighted with the return of that action. So, on average, the updates will increase the probability of those actions that had the highest return.

**Actor-critic methods.** The policy update algorithm given in Eq. (3.11) notoriously suffers from high variance and can be improved further, by using the advantage $A_\pi$ of an action $a_t$ instead of its return. It is defined as

$$A_\pi(a, s) = q_\pi(a, s) - v_\pi(s)$$

(3.12)
and evaluates if a given action performs better than the policies expected behavior. Naturally, the policy is improved by favoring more advantageous actions. However, the exact value functions \( q_\pi \) and \( v_\pi \) for the current policy are unknown and have to be approximated alongside the policy. A common choice is to approximate the state value function by another neural network \( \hat{v}_\pi(s) \). The recursive definition of the action value given in Eq. (3.5) can then be used to estimate the advantage based on \( \hat{v}_\pi(s) \) and the sampled reward. Methods that do so are called actor-critic methods, where the actor represents the policy and the critic represents the approximated value function. While those algorithms are biased by the initial value approximation, their variance is largely reduced which yields faster convergence. Hence, they are considered to be superior to plain Monte Carlo methods like Eq. (3.11), where states are independently sampled (Schulman et al. 2015b; Sutton and Barto 2018).

The optimal balance between bias and variance is subject to current research. The highest bias is produced by the one-step residual

\[
\hat{A}^{(1)}_t = \delta_t = r_t + \gamma \hat{v}_\pi(s_{t+1}) - \hat{v}_\pi(s_t),
\]

also known as Temporal Difference error, which is the difference of the actual reward and the expected reward under the current policy using the current value estimation. When taking multiple steps into account, a Generalized Advantage Estimator

\[
\hat{A}^{\text{GAE}(\gamma, \lambda)}_t = (1 - \lambda)(\hat{A}^{(1)}_t + \lambda \hat{A}^{(2)}_t + \lambda^2 \hat{A}^{(3)}_t + ...) = \sum_{l=0}^{\infty} \left( \gamma^l \lambda^l \delta_{t+l} \right)
\]

(3.14)

can be introduced (Schulman et al. 2015b). The limits of the later resemble the low variance, highly biased one step residual for \( \lambda = 0 \) and the unbiased \( G_t - \hat{v}_\pi(s) \) for \( \lambda = 1 \). Therefore the compromise between bias and variance can be controlled by the parameter \( \lambda \). The policy update is performed after recording a predetermined amount of actions, called batch, and optimizes the objective

\[
J(\theta) = \mathbb{E}_{\pi}[\hat{A}_t \log \pi(a_t | s_t, \theta)].
\]

(3.15)

Note, that when considering multiple steps the batch of actions can, but does not need to, resemble an episode. Importantly, this allows for episodes of infinite length, so the algorithm performs equally well on continuing problems like the above mentioned game of snake. If the end \( T \) of the batch does not correspond to the end of an episode, the sampled returns have to be bootstrapped by the approximate value of the last step,

\[
\hat{G}_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'} + \gamma^{T-t} \hat{v}_\pi(s_T).
\]

(3.16)

As the sampled return is also the fitting target of the approximate value function \( \hat{v}_\pi(s) \), this bootstrapping directly introduces additional bias towards the initial value estimation.
3.3 Application to multi agent systems

Clipped proximal policy optimization. When maximizing the performance defined by the general advantage estimator as given in Eq. (3.15), it is appealing to reuse the recorded trajectory for multiple steps of gradient ascent (opposed to Eq. (3.11), where only one gradient step is performed per action). However, this is “not well-justified, and empirically it often leads to destructively large policy updates” (Schulman et al. 2017). A possible solution is to use trust region policy optimization (Schulman et al. 2015a), where multiple steps are performed, but the gradient ascent is penalized by the divergence between old and new policy. This is important to ensure monotonic convergence even for highly nonlinear policy functions like artificial neural networks. A similar, but more general approach has been introduced by Schulman et al. (2017), which is clipped proximal policy optimization. It is based around the “surrogate” maximization objective

\[ L(\theta) = E_\pi \left[ \frac{\pi(a_t | s_t, \theta)}{\pi(a_t | s_t, \theta_{old})} \hat{A}_t \right], \]

originally introduced by Kakade and Langford (2002), instead of maximizing Eq. (3.15) directly. The approach is further refined by clipping the probability ratio between old and new policy to \(1 \pm \epsilon\) in order to constrain excessive policy change, yielding

\[ L_{\text{clip}}(\theta) = E_\pi \left[ \min \left( \frac{\pi(a_t | s_t, \theta)}{\pi(a_t | s_t, \theta_{old})}, \text{clip} \left( \frac{\pi(a_t | s_t, \theta)}{\pi(a_t | s_t, \theta_{old})}, 1 - \epsilon, 1 + \epsilon \right) \hat{A}_t \right) \right]. \]

This successfully limits the gradient ascent to a monotonic regime while optimizing the utilization of experience gathered in a recorded batch of actions.

3.3 Application to multi agent systems

To employ reinforcement learning in the context of collective behavior, the framework has to be expanded to multiple agents, which interact in a cooperative way. Up to this point, we have assumed a single agent, acting on the environment. Fortunately, in the case of a cooperative task, this can be expanded by the common scheme of centralized learning and decentralized execution (see e.g. Gupta et al. 2017; K. Zhang et al. 2021). Here, agents independently collect experience about the environment, but share a single policy, which is optimized by the recorded trajectories of all individuals. While the changes to the algorithms itself are minor, some important considerations have to be made when investigating multiple agents: In a single agent scenario, the response of the environment to a chosen particle action is expected to be stochastic, but constant over the time of learning. In a multi agent system on the other hand, the environment response as seen for one individual particle depends on the behavior of the other agents in the system and, therefore,
depends on the current shared policy, i.e. the environment dynamics \( p_n(r, s' | a, s) \) become policy dependent. A similar argument has to be made for the definition of the reward function: While a shared collective reward is in theory possible, it will be highly disconnected from the individual actions chosen by one agent. Subsequently, the advantages calculated for individual actions become less deterministic, which heavily degrades convergence of the policy. It is therefore advisable to keep rewards individual to the actions performed by specific agents.

Another aspect of multi agent systems and a shared policy is the representation of the system state: Contrary to the simple example of Snake with its discrete environment presented earlier, the space state for a group of agents performing collective behavior is usually continuous and high-dimensional. It becomes further complex, when the number of agents within the system is allowed to change over time. Furthermore, as the policy is shared but the action chosen by the policy has to be specific for an individual agent the input to the policy also needs to be local to that agent. Thus, it is usually provided as a finite vector of observables \( o_{it} \) local to the \( i \)th agent at time step \( t \), which is a projection of the global state. This projection can be seen as a local “perception” of the environment by an agent and can be compared to the definition of neighbors and zones in the previously discussed models of social forces.

The vector of observables serves directly as an input to artificial neural networks (ANN) which are trained to approximate the optimal policy and the corresponding value function. As discussed above, neural networks are a common choice for function approximation as they can fit nonlinear functions with arbitrary input and output dimensions without prior knowledge of their exact form. In the specific case of this thesis, we use a densely connected artificial neural network with \( N \) layers. The input layer of the network \( l_0 \) equals the observables \( o_{it} \). Each consecutive layer is then calculated as a linear transformation of the previous layer plus an offset vector, to which an activation function is applied. Hence,

\[
\begin{align*}
    l_{n+1} &= f_{\text{ReLU}} \left( A_{n,n+1} l_n + b_{n+1} \right), \\
    f_{\text{ReLU}}(x) &= \begin{cases} 
    x & (x > 0) \\
    0 & (x \leq 0)
    \end{cases} = \max(0, x),
\end{align*}
\]

where \( A_{n,n+1} \) and \( b_n \) form the weights \( \theta \) of the neural network. Activation functions can have various forms, but play a critical role in the concept as they are the main source of nonlinearity in the neural network. Here, we use a rectified linear unit (ReLU) for all intermediate layers, which is a common choice for its low cost of computation. The shape of the output layer \( l_N \) of the neural network depends on its intended use: In case of the policy network (actor), the number of
3.3 Application to multi agent systems

dimensions of the output layer have to match the number of actions available to an agent. A softmax activation function is applied to the output layer in order to form the policy as a normalized probability distribution, from which the specific action is then drawn

\[ a_{it} \sim \pi(a_{it} | o_{it}) = \frac{\exp[(lN)_a]}{\sum_a \exp[(lN)_a]} . \]  

(3.21)

In case of the value approximation network (critic), the output layer is just a scalar and directly interpreted as the value \( \hat{v}_\pi(o_i) \) without further normalization. As mentioned above, after a batch of actions has been recorded, all trajectories of individual agents can be used independently to optimize the neural networks. Throughout this thesis, this is done by clipped proximal policy optimization as described in the previous section.
Active matter is formed out of active agents which self propel. To do so, these agents constantly consume energy and, thus, perform work on the system (Ramaswamy 2010). In terms of statistical mechanics, this constant dissipation of energy consequently drives the system permanently out of thermal equilibrium. As a direct consequence, many internalized properties of equilibrium physics do not necessarily apply, for example time-reversal symmetry, which is inherently broken in active systems (Cates and Tailleur 2015).

While the concept of active matter is not bound to any length scale, here, I will focus on systems comprised of so called “microswimmers” dispersed in a (usually) liquid medium. These swimmers are active in the sense that they perform constant locomotion, but are also subject to thermal fluctuations in the form of Brownian motion due to their microscopic size (Elgeti et al. 2015). Another important factor at the micron size scale is that swimmers usually operate in an over-damped regime, where swimming concepts have to differ from the macroscopic world (Purcell 1977). While the challenges of swimming at low Reynolds numbers have been solved in a variety of ways by natural microswimmers like bacteria or sperm cells (Lauga and Powers 2009; Ishikawa 2009) more recently also artificial microswimmers with varying propulsion concepts have been created (Ebbens and Howse 2010; Bechinger et al. 2016). Research interest in microswimmers has steadily grown in the past decades where active systems have been employed as model systems to study collective behavior (Schweitzer 2007; Cates 2012; Elgeti et al. 2015; Lavergne et al. 2019), but also to design micro-robotic systems for micro-assembly or targeted transport (L. Yang and L. Zhang 2021; J. Jiang et al. 2022). Recent advances have also been reported in medical application of artificial microswimmers for drug delivery (Buea and Taboryski 2020) as well as non-pharmaceutical treatment, e.g. of thrombosis and stroke (H. Zhang et al. 2023).

This chapter will discuss common types of artificial microswimmers, the principles of active Brownian motion, dynamics in dense active systems as well as general strategies of external control.
4.1 Artificial microswimmers

Artificial microswimmers come with various shapes and propulsion methods (Bechinger et al. 2016). Pioneering the field, Paxton et al. (2004) fabricated the first particles which can be classified as artificial microswimmers: Rods of 2 µm length, consisting of gold at one end of the rod and of platinum at the other end, which self-propel when dispersed in a hydrogen peroxide solution. The propulsion is caused by the catalytic properties of the platinum end, which decomposes the hydrogen peroxide, creating an gradient of oxygen around the particle, which in return is inflicting phoretic forces on the rod. This mode of propulsion is therefore also called self-diffusiophoresis. The important underlying principle behind the self inflicted chemical gradients around the particle is its anisotropy in surface properties. This fact is equally represented in a whole family of microswimmers named “Janus particles”, which are spherical colloids which are coated with a different material on one hemisphere (also referred to as cap). These have become popular mainly for their ease of fabrication and support a variety of propulsion mechanisms: Catalytic Janus particles with a platinum cap exploit the same diffusiophoretic effect in hydrogen peroxide as described above (Howse et al. 2007). Their main disadvantage is the fact that the hydrogen peroxide is consumed over time, which leads to the particle speed slowing down with decaying hydrogen peroxide concentration. This is avoided in self-thermophoretic systems, where gold capped silica particles are propelled by a thermal gradient created by adsorbing light on one hemisphere (H.-R. Jiang et al. 2010). However, the thermal gradients and consequently the illumination strength necessary for propulsion are limiting the possible amount of microswimmers and might induce additional optical forces to the particles. Instead of being driven directly by the thermal gradient, gold or carbon capped silica spheres can also be dispersed in a binary liquid mixture which is kept close to its critical point. Here, illumination and therefore heating of the light adsorbing hemisphere leads to local demixing of the fluid and consequently self-diffusiophoretic propulsion at much lower illumination intensities (Volpe et al. 2011; Buttinoni et al. 2012; Samin and Roij 2015). This class of light-activated particles in a binary fluid, is used in the experiments described in this thesis and is presented in more detail in the following Chapter 5.

While Janus particles are a common choice for artificial microswimmers, self propulsion is not limited to self-phoretic effects. A conceptually completely different example are colloidal propellers, which have a spiral shape and include a magnetic dipole with which they are rotated by an external magnetic field (Ghosh and Fischer 2009). This results in purely hydrodynamic propulsion, mimicking the rotation of bacterial flagella. Yet another propulsion concept is employed by so called colloidal rollers, which, as the name implies, propel by rotational motion on a substrate.
4.2 Active Brownian motion

Two common types of rollers are Quincke rollers, where an angular momentum is induced in insulating colloids by an unstable electric charge distribution around the particle surface due to a strong external electric field (Bricard et al. 2013; 2015), as well as magnetic rollers, which try to synchronize their rotation to an oscillating external magnetic field (Kaiser et al. 2017; Kokot and Snezhko 2018). Contrary to colloidal propellers, where the propulsion direction is fixed by the rotation axis of the external field, rollers can move in any direction orthogonal to the external field, allowing for a wider variety of motion. While many of the Janus swimmers can in principle move unconfined through the dispersion medium, in most systems studied so far they are equally confined to a substrate due to sedimentation (and for the ease of observation with standard microscopy techniques).

4.2 Active Brownian motion

A key property of artificial microswimmers is their constant propulsion into an orientation $\hat{u}$ intrinsic to the particle (with the exception of colloidal rollers) while at the same time being subject to Brownian motion. Hence, these particles are also often referred to as active Brownian particles (ABPs), in contrast to unpropelled passive colloids. Furthermore systems of colloidal particles are usually over-damped, consequently the constant propulsion force directly translates to a constant propulsion speed $v_0$ of active particles. The equations of motions which describe such an active particle are, thus,

$$\frac{d}{dt} r = v_0 \hat{u} + \xi_0,$$

$$\frac{d}{dt} \hat{u} = \xi_r,$$

(4.1)

with independent Gaussian white noise $\xi_0$ and $\xi_r$, obeying $\langle \xi \rangle = 0, \langle \xi(t) \cdot \xi(t') \rangle = 2k_B T \delta(t - t')$. As the direction of motion is directly affected by rotational diffusion, both, translational diffusion $D_0$ and rotational diffusion $D_r = \tau_r^{-1}$, are represented in the mean square displacement of an active Brownian particle. In the case of 2-dimensional motion, it is given by

$$\langle \Delta r^2 \rangle = 4D_0 \Delta t + \frac{v_0^2 \tau_r^2}{2} \left( \frac{2 \Delta t}{\tau_r} + e^{-2 \Delta t/\tau_r} - 1 \right),$$

(4.2)

with an effective diffusion of $D_{\text{eff}} = D_0 + \frac{1}{4} \frac{v_0^2 \tau_r}{\tau_r}$ in the limit of $\Delta t \gg \tau_r$ (Howse et al. 2007). The crossover between ballistic motion and an effective Brownian random walk is identified by the persistence length $l_0 = v_0 \tau_r$ (see also Fig. 4.1). Notably, this means that while active particles are by definition out of equilibrium, their active Brownian motion can be described in terms of an effective equilibrium with effective diffusion constant and temperature. This description of active
propulsion as effective Brownian motion is also not limited to artificial active particles per se, but similar relations have also been derived for microswimmers with other propulsion patterns, like bacteria undergoing run-and-tumble motion (Tailleur and Cates 2008; Theves et al. 2015).

4.3 Dense active matter

The use of active matter as a model system for collective behavior stems from the rich and often novel phase behavior found in dense suspensions of active particles. When investigating these dense systems, an important consideration to be made in their description is the fact if hydrodynamic interactions can be neglected (so called dry active matter) or are of importance for the observed phase behavior (called wet active matter, respectively). In dry systems, only particle shape and the persistence length of active Brownian motion are relevant to the phase behavior of the system.

A common phenomena in dry active matter is the interaction with confinements, where due to the non-vanishing persistence in motion, microswimmers accumulate at walls and interfaces. Similarly, in denser systems, active particles can also spontaneously accumulate if their activity and therefore their persistence length is increased. This extensively studied phenomena is known as motility induced phase separation (MIPS) (Tailleur and Cates 2008; Buttinoni et al. 2013; Zöttl and Stark 2016, see also Fig. 4.2(A)). The impact of shape to these kind of interactions becomes apparent when investigating elongated microswimmers, which tend to align with each other for geometrical reasons. This can for example lead to the emergence of vortexes in the system (Sumino et al. 2012) and is known for large aspect ratios as active nematics (Marchetti et al. 2013). Similarly, anisotropy in microswimmer shape can also lead to anisotropic accumulation along interfaces and has for example been studied for systems where collective work is performed on a macroscopic object in a microswimmer bath (Leonardo et al. 2010).
4.4 Strategies of external control

If hydrodynamics have to be considered, a common simplification of the often complex dynamics in the vicinity of a microswimmer is provided by the squirmer model (Lighthill 1952; Zöttl and Stark 2016). It can classify microswimmers into neutral swimmers (which usually applies to e.g. Janus particles) as well as pushers and pullers, which represent common propulsion modes of bacteria and algae, but also more complex artificial microswimmers. While neutral swimmers represent a force monopole or Stokeslet and can be modeled as dry active matter in good approximation, pullers and pushers convey a force dipole, leading to attractive or repulsive forces orthogonal to the swimming direction with subsequent effects to collective dynamics (Elgeti et al. 2015; Zöttl and Stark 2016), like for example alignment with neighboring swimmers. This effect can also be observed for colloidal rollers, which similarly align due to hydrodynamic interaction, leading to global polarization or rotation in experimental systems, depending on the confinement (e.g. Bricard et al. 2013; Kokot and Snezhko 2018, see also Fig. 4.2 (B)). Another class of hydrodynamic interacting microswimmers are so called chiral particles, which perform constant rotation to varying decrees, which can lead to hydrodynamic attraction and self-assembly, which has been observed in artificial (Shen et al. 2019) as well as biological systems (Tan et al. 2022).

4.4 Strategies of external control

While intrinsic interactions between microswimmers can promote the formation of a variety of collective phenomena, other effects of collective behavior can only be modeled experimentally
with some degree of external control over the microswimmer motion. A first step towards controllable artificial microswimmers has been provided by Hong et al. (2007), who reported chemotactic behavior of catalytic rods, following a gradient of hydrogen peroxide. Similar effects of phototactic (Lozano et al. 2016) and thermotactic (Auschra et al. 2021) behavior have meanwhile also been reported for light-activated Janus particles. A big advantage of these-light activated swimmers is the possibility to apply nearly arbitrary light fields to experimental systems to control activity of particles (Buttinoni et al. 2012), which can be exploited for assembly (Söker et al. 2021) as well as for directed motion (Lozano et al. 2016; Lozano and Bechinger 2019). In combination with particle detection in an external feedback loop, specific light patterns can also be used to address individual active particles, in order to directly control their activity. This has enabled experiments of microswimmer self assembly according to rules inspired by quorum sensing (Bäuerle et al. 2018, see also Fig. 4.3) and formation of cohesive groups of active particles (Lavergne et al. 2019). As we will discuss in the following chapter, combining phototactic reorientation with individual particle control has allowed us to design a setup with previously unmatched steering capabilities in a microswimmer system as part of this thesis.
Fabrication and Control of Synthetic Microswimmers

For the experiments presented in this thesis we have used carbon-capped silica spheres, which become light-activated microswimmers when dispersed in a critical liquid mixture of water and Lutidine. At room temperature, Lutidine (2,6-Dimethylpyridine) is solvable in water but the binary mixture has a lower critical demixing point slightly above room temperature at $T_c = 34 \, ^\circ\text{C}$ and mass fraction $w_c \approx 29 \, \text{wt.\%}$ Lut. (Grattoni et al. 1993). If a Janus particle suspended in this mixture absorbs light on the carbon-coated hemisphere it can be locally heated above $T_c$, resulting in a local region of fluid-fluid demixing on the capped side of the particle, which creates a chemical gradient and induces diffusiophoretic forces that propel the particle forward (see Fig. 5.1 (A)). The hydrodynamics of this propulsion can be described in good approximation as a neutral squirmier or force monopole (Lighthill 1952). The size of our particles is comparably large, with a diameter of approximately 6.3 $\mu$m. As a consequence, when filled into a sample cell, they quickly sediment onto the lower surface due to gravitation, where they are stabilized by intrinsic negative surface charges of the silica of the particles and the quartz glass of the cell. Therefore, all active motion is confined to the horizontal plane and can be easily observed by standard microscopy techniques. Figure 5.1 (B) shows an experimental snapshot of three particles, recorded by bright-field microscopy in our setup, as done throughout all presented measurements. The edge of the carbon cap can not be fully resolved, but all particles do show an intensity gradient, which can be used to identify their orientation.

The propulsion velocity of active particles can be tuned by the intensity of illumination. As the sample temperature is usually kept below $T_c$ propulsion does not start until an offset intensity $I_0$. If the capped side is heated above $T_c$ by sufficient light absorption, local demixing occurs, propelling the particle forward. Starting at $I_0$, the velocity of particles initially increases linearly with intensity, which corresponds to a higher temperature difference on the capped side and a larger “demixing area” as sketched in the inset of Fig. 5.1 (C). If the demixing area becomes larger than the carbon cap, it expands around the particle to the uncapped side, resulting in a reversed direction of propulsion.
Figure 5.1: Light-activated self propulsion of carbon-capped particles in a binary mixture of water and lutidine. (A) Liquid-liquid phase diagram of water and lutidine. The binary mixture has a lower critical demixing point (LCP) at $T_c = 34 \, ^\circ C$ and $w_c \approx 29$ wt.\% (Grattoni et al. 1993). (B) Transmitted light microscope image of carbon-capped silica particles suspended in water-lutidine; scale bar is 5 µm. (C) Propulsion velocity of an illuminated Janus particle as a function of light intensity. Propulsion starts at an offset $I_0$ at which the absorbing cap reaches $T_c$ of the water-lutidine mixture and is proportional to intensity afterwards. The particle is propelled towards the uncapped side by a small region of demixing on the capped side (blue area in inset). For high intensities, the demixing region starts to expand around the particle, which leads to a reversal of the propulsion direction above $I_{rev}$.

and a rapid increase in velocity ($I_{rev}$ in Fig. 5.1 (C)) (Buttinoni et al. 2012; Gomez-Solano et al. 2017). For fixed levels of illumination intensity, particle velocities are assumed to be constant although in reality they can vary significantly in between particles and over time depending on surface effects and sample quality. Throughout this work, sample cells are kept at a fixed temperature of 28 °C and illumination of particles is limited to the linear propulsion regime for better reliability. The mean particle velocities achieved by this method are in the range of 0.2 µm s$^{-1}$ to 0.5 µm s$^{-1}$. In a system of passive particles only the translational diffusion is restricted to 2D by sedimentation. However, passive particles still undergo free rotational diffusion in three dimensions, which is not negligible for Janus particles as they bear an intrinsic axis of orientation. On the contrary, active propulsion fully restricts the particle to 2D due to coupling of rotational orientation and direction of translation motion (Gomez-Solano et al. 2017).

Another important property of the presented system is the phototactic behavior of particles which are subject to an intensity gradient in the illumination. If the particle orientation is not aligned with the gradient, the carbon capped is heated asymmetrically, which in turn leads to asymmetric demixing of the fluid around it. As a result, the particle experiences an active torque proportional to the sine of the difference between orientation angle and gradient direction. For a constant gradient, this results in particles reorienting towards and propelling down the applied gradient (Lozano et al. 2016).
5.1 Preparation of active particles

To prepare carbon-capped Janus particles, we obtain uncapped 6.3 μm silica spheres in an aqueous suspension from a commercial vendor (microParticles GmbH). In order to prepare the cap, a thin film of the commercial suspension is placed on a plasma-cleaned microscope slide for evaporation, leaving behind a sparse monolayer of particles. These monolayers are then coated with a 80 nm layer of carbon via carbon thread evaporation (Leica EM ACE600). After coating, the now capped particles are re-dispersed in purified water and concentrated to form a new stock suspension for sample preparation. To improve the release of particles from the microscope slide for re-dispersion, we use an ultrasonic bath. However, sonication strength has to be chosen carefully, not to release additional carbon debris from the edges of the substrate.

For experimental sample preparation, small amounts (≤ 0.1 μL) of Janus particles are drawn from the sediment of this stock suspension and dispersed in 200 μL of critical water-lutidine mixture. This dilute suspension is then filled into a plasma-cleaned 200 μm thick quartz glass flow cell (Hellma Analytics, 137-0.2-40), which is sealed with few layers of Parafilm on both ends, keeping the sample stable for several weeks (compare Fig. 5.2(A)). As the silica spheres are comparable dense (1.85 g cm$^{-3}$) compared to water and lutidin, particles rapidly sediment on the lower surface of the sample cell, where they are stabilized by the intrinsic negative surface charges of the particle’s silica and the cell. As the sample suspension is dilute, and has a gravitational length of just $l_g < 10$ nm, the particles form a quasi-2D system in very good approximation.

5.2 Feedback control for individual particle steering

In order to investigate principles of collective behavior, individual particle control is a necessity. In the presented system we implement individual control by targeted illumination of single particles with a laser beam focused to approximately particle size. While propulsion speed can be varied by the intensity of the laser beam, hitting the particle off-center or using even two beams per particle with a difference in laser intensity can be used to apply a controlled torque, therefore controlling speed and orientation of each microswimmer. To keep the relative laser position constant with respect to the moving particles, a software-based feedback loop is used to constantly track the particle positions and update the laser positions accordingly.

We use bright-field microscopy to image the sample cell during experiments. The sample itself is illuminated from the top by a blue LED (Thorlabs) as depicted in Fig. 5.2 (B) and shown schemat-
Fig. 5.2: Images of the optical setup. (A) Flow cell containing the sample, illuminated by stray laser light and the blue LED. The cell itself rests on a copper block which is kept at 28 °C by warm water circulation from a thermal bath. (B) Front view of the setup, with the camera-compartment open. From top to bottom, visible components are the imaging illumination stack including the LED, the laser illumination objective and the sample cell on its stage with the photo diode for calibration mounted in front. Hidden inside the sample stage is the imaging objective. Directly mounted to the table are the tube-lens, the color filter and the video camera. (C) Beampath leading from the laser head (left) to the AOD (red casing on the right). In-between are several polarizing beam-splitting cubes for power control and a 2:1 telescope to reduce the beam size before entering the AOD.

...cally in Fig. 5.3. The illumination is refined by a bandpass filter ($\lambda = 457.9 \pm 10.0$ nm) and a ground glass diffuser for a homogeneous and monochromatic background. Below the sample, a 20x microscope objective (Zeiss, LD Plan-Neofluar) in combination with a suitable tube-lens is used to image the sample with a CCD camera. The correction ring of the objective is set to 1 mm to match the thickness of the lower sample cell wall on which the particles sediment.

Detection and tracking of active particles is performed live during the experiments. The particles are visible on the camera image as disks with gray outline and an approximately linear gradient inside due to the carbon cap (Fig. 5.1 (B)). Video data with a resolution of 1280 × 1024 pixels is captured with 5 frames per second and transferred to a computer responsible for experiment control, where position and orientation of the particles are analyzed in a feedback control loop.
implemented in Matlab. Particle centers are calculated by a circular Hough transform with phase-coding (Atherton and Kerbyson 1999) on the gray-scale gradient of the image, adapted from the internal Matlab function `imfindcircles`. For performance improvements over the original function, format checks have been omitted and the final accuracy of particle positions has been limited to integer pixel values. After the detection of particle centers and the estimation of their radii, a small region of interest within each particle is selected to perform a principle component analysis to determine the direction of the intensity gradient and, therefore, the particle’s orientation. Labeling of single particles with unique identifiers to record their trajectories is done by a simple algorithm which is looking for the minimal distances between detected particles on adjacent video frames. As the above mentioned typical particle speed is slow compared to the 200 ms repetition time of the camera and the feedback loop, this is sufficient for reliable particle tracking. As detection errors can of course not be fully ruled out, handling of some edge cases is performed by short term memory and other techniques inspired by Crocker and Grier (1996).

Given all positions and orientations of the detected particles, i.e. the full state of the system, the propulsion which should be applied to each particle is determined. The process of doing so is specific to the kind of experiments that are performed. For experiments based on social interaction rules, these rules are directly implemented as Matlab functions and used to update the preferred direction of particles in every loop iteration. For experiments which employ reinforcement learning, the architecture more complicated. The neural networks and the training algorithm are are implemented in Python, using the popular TensorFlow software library (Abadi et al. 2016). The calculation and recording of observables and rewards (i.e. RL trajectories needed for training) are done in Python as well. A self designed network protocol is used to communicate particle positions and orientations from the Matlab control loop to the Python process and chosen actions back to the control loop.\(^1\) As the RL algorithm is relying on a meaningful change in state as a response to a chosen action to optimize the policy, a new action is only drawn every 10 s per agent while running experiments, after which the active particles have moved a considerable distance. Which action, and therefore, mode of propulsion is chosen per particle is stored in the short time memory of the live tracking algorithm within the feedback loop to be continuously applied until a new action is requested from the RL process.

---

\(^1\)The advantage of having a well defined software interface between experiment control and the implementation of the RL framework also enabled us to reuse the identical RL code for supplementary simulations, performed on the scientific compute cluster of the physics department. This has been especially useful to test and verify the correct implementation of the RL framework before deploying it to intricate experimental training.
Figure 5.3: Simplified sketch of the setup. The central part is the sample cell with the illumination objective above and the imaging objective below. As gravitation is used to confine particles into 2D, this part of the setup has to be vertically stacked. Approaching the sample from the illumination side are the imaging illumination in the form of monochromatic blue LED light and the propulsion illumination in the form of a green laser beam, which can be scanned over the sample plain. Below the sample, a colored glass filter filters out the remaining laser light while a tube lens matching the lower objective focuses the sample image on the camera.
5.3 Calibration of laser illumination

The illumination pattern to achieve the desired particle propulsion is created by rapidly scanning the laser beam over all particles with an acousto-optic deflector (AOD; AA Opto-Electronic, DTSXY-400-532). The initial beam is provided by a green cw-laser (Coherent, Verdi V2 \( \lambda = 532 \text{ nm} \)), approaching the front-aperture of the AOD with 1 mm beam width and 100 mW power, after being reduced in intensity by multiple combinations of \( \lambda/2 \)-plates and polarized beam-splitting cubes and reduced in size by a 2:1 telescope (see Fig. 5.2(C)). The AOD itself is an optical crystal into which acoustic waves are transduced to deflect the laser beam by a defined angle. With two such crystals and a computer controlled driver (input wave generator) a 2D angular space can be rapidly scanned. The diverging beams of the AOD are refocused by a telecentric 4\( f \)-telescope into the back plane of a second 20x microscope objective (Nikon, CF Plan SLWD) above the sample. Similar to optical tweezing, this second objective focuses the laser beam down to approximately particle size while translating incident angles into spatial positions within the particle plain (compare Fig. 5.3).

In our application, the scanning pattern can cover up to 200 laser positions and is repeated every 2 ms. For further details about the exact implementation of the scanning protocol, the reader is referred to Bäuerle (2020). Note, that while conceptually close to optical tweezers, no optical forces are induced on the particles as the laser intensity is very low (0.3 mW, only hitting each particle for a fraction of the time), the effectively used NA off the illumination objective is low due to the small beam diameter and the beam is deliberately focused below the sample plane. Any remaining laser light past the sample cell is filtered out before the camera with a colored glass filter.

5.3 Calibration of laser illumination

In order to propel the microswimmers in a deterministic and reliable fashion, the laser positioning and intensity requires a high level of precision, which is achieved by multiple calibration steps. The primary unit of measure is the position \((x, y)\) in the sample plain, given in micrometers. To determine the size of the field of view off the video camera, a Ronchi-ruling slide with a known pattern width is imaged instead of a sample cell. The resulting measures are saved as a calibration for use in all other image analysis, which directly returns particle positions converted to real space during experiments.

To apply a predefined laser pattern to the sample plain, a function \((x, y) \mapsto (f_x, f_y)\) needs to be calibrated, which determines the required acoustic frequencies for the AOD. To generate these mappings, two cubic polynomial functions are fitted for \(f_x(x, y)\) and \(f_y(x, y)\) for measured pairs of \((x, y, f_x, f_y)\). Measurements of such pairs are done in multiple steps: For a good first approximation for initial calibration, the color filter in front of the camera can be changed to directly image the...
position of the laser beam. To avoid any scattering, this is done with a sample cell without any particles. In a second step, a sample with uncapped silica-only particles is prepared. The usual image analysis tools are used to determine the center of a randomly chosen uncapped particle. Following this, a small grid of laser beams is scanned over the particle position (using the initial calibration). Scattering light from the particle is recorded by a photo diode next to the sample and used to determine the frequency of strongest back-scattering by a Gaussian fit to the scanning grid. This point is assumed to be the particle position in \((f_x, f_y)\)-space, although it might have a constant offset due to the angle of the photo diode to the sample. Contrary to the protocol described in Bäuerle (2020), this process is repeated for several hours choosing a random particle each time. Through diffusion of particles within the sample plain, a high amount of statistic can be collected in such a way to obtain optimal input for the polynomial fit. The initial calibration data by direct laser imaging is discarded when fitting the refined data obtained by scattering.

A second calibration function is necessary to determine the required amplitude \(A_f\) of the acoustic wave to obtain a laser spot with desired power \(P\). As the defraction efficiency of the AOD depends on the defraction angle, the amplitude also depends on the applied frequencies. For simplicity, it proved to be sufficient to decouple the necessary factors into \(A_f = g_1(P \cdot g_2(f_x, f_y))\), where \(g_1\) is a fourth order scalar polynomial (Bäuerle 2020) and \(g_2\) is a two-dimensional cubic spline interpolation. Experimental data to construct \(g_1\) and \(g_2\) is obtained by placing a photo diode directly into the beam path instead of the sample cell. Combining the two independent calibrations, a full mapping \((x, y, P) \mapsto (f_x, f_y, A_f)\) is obtained to drive the AOD.

### 5.4 Implementation details of propulsion and rotation

As noted before, individual particles are propelled by targeted laser beams. Simply positioning the laser on the particle center will heat the cap symmetrically and therefore lead to active Brownian motion of the particle, comparable to a flat illumination of the whole sample. However, in order to additionally induce an active torque to reorient a particle, its cap needs to be illuminated asymmetrically. For global illumination of particles with a light gradient, a reorientation rate of

\[
\omega = \omega_{\text{max}} \sin(\gamma)
\]

has been found previously (Lozano et al. 2016), where \(\gamma\) is the angle between particle orientation \(\hat{u}\) and the direction of the light gradient. We found this relation to be approximately true for a single laser spot as well, if its position is shifted away from the particle center by a fixed offset. This will create an intensity gradient on top of the particle and reorient it to a specified target direction \(d\).
5.4 Implementation details of propulsion and rotation

**Figure 5.4**: Particle reorientation. By offsetting the laser spot away from the particle center, a light gradient is created on top of the particle, which induces a torque reorienting the particle away from the laser spot (towards the target direction \( \boldsymbol{d} \)). For large angles \( \gamma \) between particle orientation \( \hat{\boldsymbol{u}} \) and target direction, the particle is not only reoriented, but will also move side-wards away from the laser spot. This can be counteracted by a second, weak laser spot in order to utilize the maximum rotation rate \( \omega_{\text{max}} \) at \( \gamma = 90^\circ \).

Opposite of the offset, as seen in Fig. 5.4 (top). Using this technique, \( \omega_{\text{max}} \) depends on the laser offset and the beam width as they define the strength of the gradient, but does not necessarily in a linear way. Furthermore, while larger offsets correspond to a higher torque in general, they also lead to a side-wards “drifting” motion of the particle away from the laser spot, especially for large angles of \( \gamma \). An offset of 1.8 \( \mu \text{m} \) has been found to achieve \( \omega_{\text{max}} \approx 4^\circ \text{s}^{-1} \) (Bäuerle et al. 2020) while not imposing too much drift for usual angles of \( \gamma \) and was used in the experiments on social interaction rules described in Chapter 6. Additionally, a larger offset of 2.5 \( \mu \text{m} \) was used as well to “push” particles around without regard for their orientation. This was mainly used to keep new particles out of the measurement area and to reset particles to initial positions in-between measurements.

In order to achieve a consistently high rotation rate without imposing drift on the particles, a technique including two laser spots per particle has been developed for the reinforcement learning experiments discussed in Chapters 7 and 8: A main laser spot of normal intensity is set with an offset of 1.8 \( \mu \text{m} \) to one side of the particle (corresponding to \( \gamma = 90^\circ \)) as seen in Fig. 5.4 (bottom). A second, weaker laser spot is set to the opposite side of the particle. Crucially, this second spot has been found to counteract the side-wards drifting motion well, while not reducing the rotation rate by a relevant amount. Note, that while the particle is not moving side-wards with this configuration, it is still propelling forward towards its intrinsic direction \( \hat{\boldsymbol{u}} \), which results in a minimal turning radius of approximately 10 \( \mu \text{m} \) using this approach. As all laser spots are part of a rapidly scanned sequence, the time delay between the two spots is an important quantity as well. The best effect on particle motion has been found for the secondary weak spot exactly preceding the stronger main spot in the scanning sequence.
Swirls of Active Particles based on Social Interaction Rules

The ability to propel and reorient individual particles allows us to model a wide variety of collective behavior. The main limitation of our experiment is the finite field of view, which covers an area of 400 $\mu$m × 320 $\mu$m. Observations of collective group dynamics of active particles are therefore limited to swarming and swirling, as any flocking group with a reasonable persistence length in traveling would quickly reach the edge of the measurement area. Unordered swarms and rotating swirls on the other hand can both be considered as cohesive but stationary configurations (Chapter 2). Their key difference is their degree of internal order: A swarm has no order of particle orientations and can be modeled by pure social attraction (and possibly short range repulsion). A swirling motion on the contrary additionally requires some mechanism of alignment between the individual agents of the group.

Previous studies have shown that a certain balance of attraction and alignment forces needs to be present in order to maintain stable swirling motion (Couzin et al. 2002; Delcourt et al. 2016). Pure attraction will likely result in an unordered swarm, while pure alignment is optimally satisfied in a parallel flock. On a local level, the balance between alignment and confinement can be broken down to two favored directions of a single group member: If the particle is situated in an already established swirl, an alignment force will favor a tangential orientation of individuals along the direction of rotation. An attractive force on the other hand will favor a radial alignment towards the center of the group and therefore the center of the swirl. If a social interaction rule is imposed, where the balance between those forces is given as a free parameter, it should be possible to observe both unordered and ordered group states, depending on the bias towards either attraction or alignment.

Here, we impose such an interaction rule on our experimental system of active particles which indeed leads to unordered swarms as well as emerging swirling motion depending on the choice of a single parameter. As we can change this parameter continuously, we are also able to investigate
the nature of the underlying transition between the two states of collective motion. Notably, our results show a clear bifurcation point in the recorded rotational order of the particle group, which we can describe by a simple Landau-like model. The observed criticality is remarkable in the context of collective behavior, as it is a common theory that the collective dynamics of many groups are close to a critical transition for advantages such as increased response to external cues and scale-free behavior within the group (Chapter 2).

6.1 Interaction rule

The particular social interaction rule imposed on our system is designed to cope with the physical constraints of the experiment: Active particles are extended bodies so, contrary to many theoretical models, there is an intrinsic hard body repulsion between single agents which becomes increasingly significant with increased particle density. Furthermore, while particles are able to actively reorient, they have a maximum angular velocity and are not able to “instantly” adjust their orientation, as it is for example assumed in the Vicsek model. In fact, we are not able to apply an interaction rule that directly dictates the motion of particles. Instead, a preferred heading direction \( \hat{d}_i \) is calculated individually, towards which the \( i \)th agent would like to move. Due to the finite rotation rate the actual particle orientation \( \hat{u}_i \) will only follow the preferred heading direction with a certain delay. Similar, active particles are assumed to have a constant propulsion speed, but their displacement within one time step is not only subject to noise and particle polydispersity but also to all possible interactions with other particles, which includes steric as well as hydrodynamic and phoretic interactions.

In order to achieve a swirling motion based on the preferred particle direction \( \hat{d}_i \), it needs to reflect the necessary balance between attraction to the group and local alignment with neighbors. If only attraction is considered (and each agent has unlimited information about its environment), the heading direction would strictly point to the center of mass (COM) of the particle group. On the other hand, if only local alignment would be considered the heading direction would be strictly tangential to a preexisting swirl, leading to a loss of cohesion and eventually some Vicsek-like behavior. To ensure stable swirling in noisy environments, we have fixed this balance geometrically such that the preferred heading direction of an individual particle always deviates by a given angle \( \Delta \in [0°, 90°] \) from the direction to the center of mass. While the magnitude of this angle \( \Delta \) is set as the main parameter of the interaction rule, its sign is chosen individually for each particle \( i \) such that the particle orientation \( \hat{u}_i \) is best aligned with the mean orientation \( \langle \hat{u} \rangle_i \) of its nearest neighbors (shown for the red particle in Fig. 6.1). Relevant neighbors are selected based on a distance
threshold \( R_{\text{align}} = 25 \, \mu\text{m} \), indicated by the blue circular area. The only exception to this rule is when neighboring particles are closer than a threshold \( R_{\text{rep}} = 8 \, \mu\text{m} \), which corresponds to a clearance of about \( 0.25 \sigma \) between particles, for which a short range collision avoidance is implemented, which turns the particles away from each other to avoid direct contact. We expect that given a certain magnitude of angular deviation, particles will eventually converge to one direction and form a collectively swirling group.

### 6.2 Collective states

Indeed, if the angular deviation \( \Delta \) is chosen sufficiently large, stable swirling motion is observed in experiments. On the other hand, if \( \Delta \) is close to zero, there is no rotational order present and the system can be described as an unordered swarm. Figure 6.2 displays images of 50 experimental microswimmers while being subject to the interaction rule in three scenarios with different parameter \( \Delta \). If \( \Delta \) is set to zero, i.e. all agents are trying to head to the center of mass, the system rests in a mostly jammed swarm state with no rotational order. For a small angle of \( \Delta = 22.5^\circ \) the system is less jammed and there are larger fluctuations in particle motion, however, no global rotational order is observed. For \( \Delta = 45^\circ \) on the other hand, the trajectories of particle motion plotted in the background of Fig. 6.2 clearly indicate a collective swirling motion.

In order to quantify the strength of rotational motion, we introduce the rotational order parameter

\[
O_R = \frac{1}{N} \sum_{i=1}^{N} (\hat{r}_i \times \hat{u}_i) \cdot \hat{e}_z ,
\]

(6.1)

where \( N \) is the number of particles, \( \hat{r}_i \) is a unit vector pointing from the center of mass to the \( i \)th particle, \( \hat{u}_i \) is the unit vector denoting the particle orientation and \( \hat{e}_z \) is the unit vector perpendicular to the sample plain. As a result, \( O_R \) represents the average tangential alignment of particles with
6 Swirls of Active Particles based on Social Interaction Rules

Figure 6.2: Experimental snapshots of $N = 50$ active particles, subject to our fixed interaction rule with different values of angular deviation $\Delta$. Particles are recorded by digital video microscopy, and annotated in the background with particle trajectories of the last 10 min. While the particles in the first two snapshots perform random swarming motion, the third snapshot displays clear signs of a collective swirl in counter-clockwise direction.

the group and therefore the global rotational order from fully counter-clockwise ($O_R = +1$) to clockwise ($O_R = -1$). Figure 6.3 (A) shows the time evolution of $O_R$ for two experimental instances with $\Delta = 0^\circ$ (black line) and $\Delta = 67.5^\circ$ (green line). The difference in absolute rotational order, or more precisely the lack thereof in the first case, is clearly visible in the graph. In fact, when measuring the time average of absolute rotational order $\langle |O_R| \rangle$ over a broad range of $\Delta$ as shown in Fig. 6.3 (B) a clear dependence is visible. Note that due to the small system size $\langle |O_R| \rangle$ does not completely vanish for small $\Delta$, as fluctuations in particle orientations are strong in comparison. This is displayed with more emphasize in Fig. 6.3 (C) where the probability distribution $P(|O_R|)$ is shown for selected values of $\Delta$. While for $\Delta = 22.5^\circ$ the distribution is clearly broadened, the peak remains at zero, so the system can still be assumed to be in a swarm state. For higher values of the control parameter $\Delta$ the peak of the distribution is shifted towards higher values of $|O_R|$, indicating a continuous transition from a swarm to a swirl. The presence of a well defined transition between swarm and swirl is further supported by symmetry arguments: While the swarm state has no internal order, symmetry is broken for the swirl state into either clockwise or counter-clockwise motion.

While rotational symmetry is broken for the swirl state, due to the finite system size and strong fluctuations the average rotational order remains $\langle O_R \rangle = 0$ for sufficiently large times, as even for large values of angular deviation $\Delta$ spontaneous reversal of the direction of rotation can occur. Because the direction of motion is a binary decision for every individual agent, coupled to the mean orientation of its neighbors, the group represents dynamics similar to an Ising system. As such, in
6.2 Collective states

Figure 6.3: Rotational order parameter in dependence of deviation angle \( \Delta \). (A) Time evolution of \( O_R \) for \( \Delta = 0^\circ \) (black line) and \( \Delta = 67.5^\circ \) (green line). For \( t < 0 \) the system is in an artificially prepared initial state, hence no rotational order is present in any case. (B) Time average of absolute \( |O_R| \), individually measured for a different values of the control parameter \( \Delta \). (C) Probability distribution (over time) of \( |O_R| \) for selected values of \( \Delta \). The mean of the displayed distributions correspond to the respective value plotted in (B).

the presence of any fluctuations (nonzero “temperature”) the sense of rotation would only be stable in the limit of infinite system size. Figure 6.4 (A) displays two events where the rotational order of the system is lost due to fluctuations. One short event, where swirling order is reestablished with the same sense of rotation and one longer event where the sense of rotation is reversed to negative \( O_R \). To reveal the internal structure of this transition from counter-clockwise to clockwise rotation, we can also calculate a locally resolved \( O_R \) using a weighted average approach. Panels b-g show the color coded local \( O_R \) for consecutive snapshots of the system state, as marked in Fig. 6.4 (A).

Remarkably, during the transition, local order to individual particles is never completely lost, but reversal of direction starts as a fluctuation on one side of the swirl and is propagated as a growing “domain” until it becomes the major new direction of rotation. To quantify that this not a coincident but the general process when the rotation reverses, we can evaluate a second order parameter for global polarization of the group,

\[
O_p = \left| \frac{1}{N} \sum_{i=1}^{N} \hat{u}_i \right|.
\]

If \( O_p \) is high (close to 1), the group is aligned into a common swimming direction. When plotting the combined probability distribution for observations of any given combination of \( O_R \) and \( O_P \) (see Fig. 6.4 (H)), we can see that agents always keep a certain degree of local alignment and the transition between two directions of rotation always passes an intermediate state of increased polarity. This intermediate state represents the configuration in which two halves of the group with opposing directions of rotation form an aligned flock, as seen in Fig. 6.4 (D,E).

In fact, when studying the order parameter histogram in Fig. 6.4 (H), the transitions between clock-
Figure 6.4: Spontaneous reversal of direction of rotation for measurements recorded with a deviation angle of $\Delta = 67.5^\circ$. (A) Time evolution of a reversal event within a single measurement. Markers indicate the instances at which snapshots have been taken. (B-G) Snapshots of the experimental system, color-coded by a localized version of the rotational order parameter $O_R$. Over the course of the reversal event, two “domains” with high counter-clockwise ($O_R = +1$) and clockwise ($O_R = -1$) can be observed. (H) Histogram over multiple measurements counting the observation probability for combinations of order parameters $O_R$ and $O_P$. While the two swirling states are by far the most probable configuration to find the system in, a small intermediate peak can be distinguished for a flock-like polarized state in between.

wise and counter-clockwise swirls seem to be not completely instantaneous but this intermediate flock can be identified as an independent peak, representing a third although far less stable dynamical state of the group. This observation is not covered by the simplified Ising-like description proposed above, however, small adjustments to the rule can lead to more emphasis on this flocking state. Most notably, this effect can be seen in the work of Chen and Bechinger (2022), where an external thread source is introduced which leads to an escape motion of the group exploiting this polarized configuration.

6.3 Theoretical model for continuous transition

As discussed above, due to the finite system size the time averages of neither rotational order $O_R$ directly, nor $|O_R|$ are suitable order parameters to identify a transition angle $\Delta_t$ between swarm and swirl. In fact, due to the inherent out-of-equilibrium nature of active particles it is not even clear a priori if the system can be described in terms of equilibrium statistics. However, we can model the time evolution of the order parameter $O_R$ based purely on symmetry arguments. As a general stochastic quantity, it has odd time reversal symmetry in the absence of any symmetry-breaking...
6.3 Theoretical model for continuous transition

external field. Thus, an equation of motion can be written as

\[ \frac{\partial}{\partial t} O_R = -aO_R - bO_R^3 + \mathcal{O}(O_R^5) + \eta(t) , \]  

(6.3)

where \( \eta(t) \) is approximated as Gaussian white noise, i.e. uncorrelated with zero mean \( \langle \eta \rangle = 0 \). All other terms have been added permissible by their symmetries (Kardar 2007; Löffler et al. 2021). Signs have been chosen such that

\[ \frac{\partial O_R}{\partial t} = -\frac{\partial V(O_R)}{\partial O_R} + \eta(t) \]  

(6.4)

resembles a Landau-like potential

\[ V(O_R) = \frac{a}{2} O_R^2 + \frac{b}{4} O_R^4 . \]  

(6.5)

To fit this model to the our data, experimental trajectories (which already come in discrete time steps) were sampled into discrete bins of \( O_R \) for which \( \langle \partial/\partial t O_R \rangle \) has been calculated based on the change in \( O_R \) to the next time step. The bin-counts were used as weights to the fitting function, to equally consider all time steps in the trajectories. As seen in the selected fits shown in Fig. 6.5(A), the model matches the experimental data surprisingly well. When plotting all fit parameters \( a \) and \( b \) over the measured values of \( \Delta \) (compare Fig. 6.5(B,C)), we can see that while parameter \( b \) is constant in first approximation, parameter \( a \) shows a linear dependency to \( \Delta \) in the vicinity of its zero transition. A direct consequence of this transition is the existence of a critical bifurcation point in the steady state order parameter \( \langle O_R \rangle^* = \pm \text{Re} \left( \sqrt{-a/b} \right) \) as shown in Fig. 6.5(D). For our data this bifurcation is found at \( \Delta_c \approx 28^\circ \) and marks the transition between the two collective dynamic states of unordered swarm and symmetry broken swirl. While the chosen approach is the most simplest stochastic model to describe the non-trivial bifurcation dynamics of the rotational order parameter \( O_R \), it captures the observed phenomena for a broad range of deviation angles. The linear relation \( a \propto \Delta - \Delta_c \) only breaks at large angles \( \Delta > 45^\circ \) where \( a(\Delta) \) saturates due to the limits of \( O_R \in [-1, 1] \). Mapping this regime to the Landau-like potential as well would require the inclusion of higher order terms.

A different aspect of the order parameter dynamics which is a direct consequence of the existence of a bifurcation point is a vanishing of restoring force for fluctuations in the order parameter close to the bifurcation. This can be identified as the divergence of a timescale \( \tau = a^{-1} \) in the auto correlation of the time evolution of the order parameter and is generally referred to as “critical slowing down”. As a indicator for the presence of a bifurcation or tipping point it is used to study complex dynamical systems over a variety of fields (Scheffer et al. 2012) and has previously been reported for natural swarms of midges (Cavagna et al. 2017).
Figure 6.5: Fit to minimal theoretical model. (A) Fit of \( \langle \partial / \partial t O_R \rangle = -aO_R - bO_R^3 \) to binned experimental data (markers). The gray-scale intensity of the markers indicates observation probability and is used as a weight to the fit. (B,C) Fit parameters \( a \) and \( b \) for different angles \( \Delta \). Markers represent the fit-parameters as obtained for the red lines in (A). Close to the bifurcation point, \( a \) resembles linear behavior with \( \Delta \), while \( b \) is in first approximation constant (blue lines). (D) Bifurcation in \( \langle O_R \rangle^* = \pm \text{Re}(\sqrt{-a/b}) \), using the data and linear dependence from (B,C). For high values of \( \Delta \), \( O_R \) is limited to \( \pm 1 \) which is not captured by the minimal model of \( a \propto \Delta - \Delta_c \).

Figure 6.6: Critical slowing down near the critical point. (A) Normalized auto correlation of rotational order parameter \( C(O_R) = \langle O_R(t) \cdot O_R(t + \Delta t) \rangle \) for increasing angles \( \Delta \) (solid lines) with bi-exponential fits \( A_1 \exp(-\Delta t/\tau_1) + A_2 \exp(-\Delta t/\tau_2) \) (dashed lines), where \( A_1, A_2, \tau_1 \) and \( \tau_2 \) are free fitting parameters. (B) Dependency of the fitted timescales \( \tau_1 \) and \( \tau_2 \) to the increasing deviation angle. The color coding represents the respective fitted amplitudes \( A_1 \) and \( A_2 \), the error bars correspond to the 95% confidence interval of the fitted timescales.
6.4 Hysteresis in behavior

In our analysis of the auto correlation of the rotational order parameter $C(O_R)$ we actually find two distinct time scales $\tau_1$ and $\tau_2$. These are identified by applying a double exponential fit to $C(O_R)$ as displayed in Fig. 6.6 (A). The short timescale of $\tau_1 \approx 10$ s is approximately constant over the evaluated range of $\Delta$ and can likely be attributed to single particle dynamics. At small angles $\Delta \approx 0^\circ$ these are mainly governed by the attraction to the group center and the short range particle-particle repulsion, implemented into the interaction rule. However, close to the critical point a second time scale $\tau_2$ emerges, rising from 200 s up to 1800 s and clearly dominating the dynamics near $\Delta_c$. This is clear evidence for critical slowing down, where the relaxation of order parameter correlation is significantly affected by the proximity to the critical point.

6.4 Hysteresis in behavior

Given the remarkable agreement of our observations with the Landau-like model, we can investigate a further phenomenon usually connected to equilibrium phase transitions: Hysteresis in the response to an external field. In our model as described in Section 6.1, the binary decisions of which direction to choose relative to the center of mass depends on the mean orientation of neighbors $\langle \hat{u} \rangle_i$. We can slightly modify this to $\langle \tilde{u} \rangle_i = \langle \hat{u} \rangle_i + h (e_z \times \hat{r})$, where $h$ is the strength of the symmetry breaking external field, which adds a bias in tangential direction to the mean neighbor orientation. Equivalent to the order parameter, $h > 0$ corresponds to a counter-clockwise preference and $h < 0$ to a clockwise preference, respectively.

To record comparable hysteresis measures, a fixed 20 min protocol was established in which swirls have been prepared with $h_{\text{initial}} = \pm \infty$ for 5 min, after which $h$ has been changed instantaneously to a finite value. To account for finite reaction time in particle motion, the system has been given 2 min to relax, after which the average $O_R$ of the remaining 13 min of measurement time has been recorded. An example $O_R$ trajectory under this protocol can be seen in Fig. 6.7, where the sample has been prepared with $h_{\text{initial}} = +\infty$ and the thicker light blue line indicates the final time average defined as $O_R(h = -0.11)$ for $\Delta = 45^\circ$. To increase statistics, the same protocol has been repeated multiple times for all combinations of $\Delta$, $h_{\text{initial}}$ and $h$ (final). All measurements taken by this protocol are displayed as data points in Fig. 6.8 (A), where blue markers correspond to positive and red markers to negative values of $h_{\text{initial}}$, respectively.

As expected, for $\Delta \leq \Delta_c$ we do not observe a dependence of the final $O_R$ to the initial state, while for $\Delta > \Delta_c$ a pronounced hysteresis loop can be seen. In lack of a more specific theoretical model,
6 Swirls of Active Particles based on Social Interaction Rules

Figure 6.7: Example $O_R$ trajectory of a hysteresis measurement under the described protocol for $h_{\text{initial}}(t < 0) = +\infty$ to $h(t > 0) = -0.11$ for a swirl at $\Delta = 45^\circ$. The swirl is prepared for 5 min, relaxed to the new value of $h$ for 2 min and the average $O_R$ (thick blue line) recorded for the remaining 13 min.

Figure 6.8: Direct hysteresis and susceptibility measurements. (A) Response of $O_R$ to an external field with strength $h$ for different values of interaction rule parameter $\Delta$. Each marker indicates an individual measurement where the system has been prepared with $h = +\infty$ (blue) or $h = -\infty$ (red) and then suddenly relaxed to constant finite value of $h$. The plotted $\langle O_R \rangle$ corresponds to a 13 min time average after an initial relaxation time of 2 min. Filled markers correspond to points which have been taken into account to fit Eq. (6.6) (solid line). Dashed lines are guide to the eyes to indicate the jumps in the hysteresis of $O_R$. While the final state of the system is independent from the initial state for $\Delta < \Delta_c$, hysteresis is observed for $\Delta > \Delta_c$. (B) Susceptibility $\chi$ evaluated from the slopes of the fitted curves in (A) at $h = 0$. The error (gray band) corresponds to the 70% confidence interval of fit parameters $A$ and $B$. 
the data has been fitted with a function of the form of the mean-field solution of the Ising model

\[ O_R(h) = A \tanh(BO_R + Ch) , \]  

which describes the data well. As we expect the noise of the system to be independent of the deviation angle \( \Delta \), the scaling parameter \( C \) of the external field, which corresponds to the curvature of the function, should be the same in all fits and has been fixed to \( C = 8 \), yielding best agreement for all curves. Parameters \( A \) and \( B \) are free fitting parameters and both correspond to the saturation of \( O_R \) for \( h \to \pm \infty \) as well as the depth of the hysteresis loop.

Note, that in any Ising system with finite noise hysteresis is always a dynamic phenomenon. As described earlier, instantaneous reversals are always possible and, therefore, in the limit of \( t \to \infty \) the area under the hysteresis loop will always vanish. Similarly, the mean field solution fails to describe the dynamics beyond the turning point of the model. This has been taken into account by only considering data points were \( \langle O_R \rangle(h) > 0 \) for \( h_{\text{initial}} > 0 \) and \( \langle O_R \rangle(h) < 0 \) for \( h_{\text{initial}} < 0 \) respectively. Data points \( O_R^{\text{data}} \) satisfying this criterion are displayed as filled markers in Fig. 6.8 (A), while neglected data points are shown as empty markers. Fitting of the model has been performed by numerically solving for

\[ \min_{A,B} \sum |A \tanh(BO_R^{\text{data}} + Ch^{\text{data}}) - O_R^{\text{data}}|^2 . \]  

As a direct consequence of the existence of the hysteresis curve, we can also give a direct susceptibility measurement

\[ \chi := \frac{dO_R}{dh} \bigg|_{h=0} . \]  

With the fitted parameters \( A, B \) at hand, we can obtain the susceptibility as the derivative of the hysteresis curve

\[ \chi = \frac{2AC}{1 - 2AB + \cosh(2BO_R,0)} , \]  

where \( O_{R,0} \) is the remanence value, obtained from numerically solving for

\[ O_{R,0} = A \tanh(BO_{R,0}) . \]  

As shown in Fig. 6.8 (B), the susceptibility has a clear maximum in agreement with the bifurcation point found in the previous section. Note, that we do not rely on the fluctuation-dissipation theorem and equilibrium properties but provide a direct response measurement to obtain the susceptibility. This is possible by the control we have other the interaction rule which is not possible in living systems and in contrast to previous work on this field (Calovi et al. 2015).
6.5 Discussion

We have imposed a social interaction rule on our experimental system of active particles, which lead to the formation of unordered, but cohesive swarms as well as rotationally ordered swirls depending on the choice of a deviation angle $\Delta$. The emergent behavior proved to be robust not only against thermal noise but also other complexities of the experimental environment. These include heterogeneities in particles and their response to the steering mechanism as well as complex physical interactions between particles based on hydrodynamic and phoretic effects. Such effects are not easy to quantify and hard to capture in simulations, but often relevant in applications.

Another remarkable feature of the interaction rule investigated in this chapter is the critical transition between the collective states of swarm and swirl. Continuously varying the angular deviation leads to a clear bifurcation in the rotational order parameter, which we were able to describe very well by a simple coarse-grained model. Despite the finite size of the system with just 50 particles, typical effects of a critical transition like critical slowing down and extensive hysteresis in the direction of rotation have been observed. This demonstrates that several advantages that a biological system might have from operating close to a “critical point” remain and can be quantified if the system size is limited and a field-theoretic approach like the model of Toner and Tu (1995) is not necessarily applicable.

While the focus of this chapter has been laid on the angular deviation angle, other parameters of the proposed interaction rule of course also influence the emerging collective behavior. These include the radii of repulsion and alignment as well as a possibly limited vision cone and interactions with other features of the environment. In fact, follow up research by Chen and Bechinger (2022) showed the emergence a flocking escape motion based on the same interaction rule, which is similarly robust and even persists if a considerable part of the group is unaware of the trigger.
The concept of reinforcement learning gives rise to a new approach in modeling collective behavior. In contrast to social forces which determine the instantaneous reaction of an agent, the idea of a reward enables the definition of higher order goals. While the spectrum of possible ways to reward an agent is generally broad, in the following chapter we want to focus on a reward which is a deterministic function of the local state of an agent, i.e. its position relative to the group of other particles, and independent of the previously chosen action. In the context of collective behavior this kind of reward function could be described as a “social potential” which includes a optimal position for the individual agent and from which social forces could be derived.

Here, we will try to recreate the swirling motion described in the previous chapter based on the framework of multi-agent reinforcement learning, in order to understand similarities but also key differences between the two approaches. To achieve this, the reward function includes two typical aspects of collective behavior: Aggregation into a cohesive group as well as collision avoidance. Positive reward is given to agents for general proximity to neighboring particles. This should lead to attractive behavior and aggregation of all particles into one group in order to optimize the reward for every individual. Strong negative reward is given to any agent in close proximity or direct contact to any other particle in order to enforce collision avoidance between active particles. While attraction to other group members is a common feature in all kind of living systems, collision avoidance is more important to some species than to others. The obvious examples are schools of fish and flocks of birds, where individual maneuverability would be highly impacted by direct contact with other individuals. The solution to this problem most often observed in nature is to align with neighboring group members (Parrish et al. 2002; Couzin and Krause 2003) in order to minimize potential trajectory overlaps and therefore potential collisions with other group members and has recently also be shown to be the best strategy in theoretical analysis (Borra et al. 2021). Combining the aspects of attraction and alignment is expected to lead to a swirling motion, as discussed before (Delcourt et al. 2016).
7 Swirls of Active Particles based on Reinforcement Learning

7.1 Implementation of reinforcement learning

As described earlier, the underlying Markov decision problem of a reinforcement learning task is defined by the interplay between the policy $\pi(a | s)$ and the environment dynamics $p(r, s' | a, s)$. The policy decides on an action to choose, given the current state of the environment, while the environment dynamics represents the resulting change to the environment. In our case of training a system of microswimmers, the environment response is fully governed by the physical experiments, where the state $s$ is defined by the recorded configuration of particle positions and orientations in the sample cell. In this chapter, we look at $N = 30$ active particles with individual rewards, which behave according to and are trained to improve a shared central policy, which is defined as an artificial neural network with three intermediate layers with 32, 16 and 16 dimensions each. That is, there is only one policy, but individual actions are drawn for every particle given its local state in form of a set of observables. Trajectories of actions, states and rewards are recorded per particle and used to improve the central policy by clipped proximal policy optimization with a generalized advantage estimator. As this is an actor-critic algorithm, a neural network for state value estimation with the same configuration of intermediate layers is trained alongside the policy. Notable parameters used for training are $\gamma = 0.97$, $\lambda = 0.97$, $\epsilon_{\text{clip}} = 0.07$, a learning rate of $\alpha = 0.003$ for both networks and a maximum of 50 epochs in policy gradient ascent, see also Chapter 3 for more details. Both networks are initialized with random weights $\theta$ prior to any training.

The actions available to an individual particle are limited to different modes of propulsion. Namely, there is a standard forward motion, as well as rotating motions to either the left or the right hand side. While forward motion is experimentally implemented with a single laser spot, rotation in either direction is performed by two laser spots per particle. These laser positioning can be seen in Fig. 7.1 and is described in more detail in Section 5.4. Note, however, that the action specific propulsion pattern is no longer applied if a particle is getting close to the boundary of the measurement area. This can happen especially often in early training, when particles basically perform random motion. In this case, a deterministic protocol takes over the control of the particle in question, which is then reoriented until it is facing towards the center of the measurement area, providing a effectively reflective boundary condition. To make this process transparent to the RL

![Figure 7.1: Possible actions available for a microswimmer: Forward motion (blue), implemented with one experimental laser spot, or left (red) or right (green) turning motion, implemented with two laser spots each.](image-url)
7.1 Implementation of reinforcement learning

Figure 7.2: Conceptual representation of the RL decision process. (A) Each agent is assumed to have a 180° forward vision divided in five sections $m = 1 \ldots 5$, for which observables for neighboring particles (lime green) and their orientations (purple) are calculated. The gray areas correspond to the blind angle and shadowed regions for which the reference particle has no vision. (B) The calculated observables serve as input to the artificial neural network which forms the policy. The output comprises a probability distribution of possible actions, from which the actual next move of the particle shown in (A) is drawn.

framework, trajectories are cut into independent parts before and after the reorientation and the time during reorientation is discarded from the training process.

The observables which describe the local state of an particle, can be interpreted as the “perception” of the environment by that agent. Inspired by other models of collective behavior, these are based around the concept of vision and represent a 180° forward facing vision cone, divided in five discrete sections $m = 1 \ldots 5$, depicted in Fig. 7.2 (A). For each section, the agent is able to observe the mean density neighboring particles as well as their mean orientation, each of which are weighted by the inverse distance between the agent and the other particles. As orientation is a directorial quantity, this leads to a total of 15 observables for the $i$th agent at time $t$, defined as

\[ p_{itm}^d = \sum_j \frac{\sigma}{|r_i - r_j|} g_{ijtm}, \]  
\[ p_{itm}^{\sin} = \sum_j \sin(\theta_i - \theta_j) \frac{\sigma}{|r_i - r_j|} g_{ijtm}, \]  
\[ p_{itm}^{\cos} = \sum_j \cos(\theta_i - \theta_j) \frac{\sigma}{|r_i - r_j|} g_{ijtm}, \]

where $r_i$ is the particle position, $\theta_i$ is the particle orientation and $g_{ijtm} \in [0, 1]$ is the fraction of particle $j$ visible to particle $i$ in section $m$. The latter is important, as the particles in the system are considered “dense” and block the vision of an agent, as indicated by the gray shadows in Fig. 7.2 (A). Thus, only information about neighboring particles in direct sight is included in the observables. Note that, due to the $1/r$ dependence and the way vision shadowing is calculated, the values of
7 Swirls of Active Particles based on Reinforcement Learning

the observables $p_d$ correspond approximately to the fraction of the horizon which is covered by neighboring particles. The strength of the resulting observables for the specific configuration of neighboring particles shown in Fig. 7.2 (A) is displayed in Fig. 7.2 (B) as inputs of the neural network forming the policy. The output of the policy is a probability distribution from which then one of the three actions is drawn.

The formal implementation of the reward function is inspired by common physical potentials and is given by

$$r_{ij}(s_i, t) = \sum_j \sigma |r_i - r_j| - 3 \sum_j \sigma |r_i - r_j|^3,$$

(7.4)

where $\sigma$ is the particle diameter and $r_{ij}$ are the particle positions. The positive reward for general proximity has a long range scaling of $1/r$, similar to the contribution of neighbors to the input observables. Notably, this means that for any agent approaching the group from a distance, the reward will grow proportionally with the observables $p_d$ for neighbor density. However, within the group, reward is not necessarily correlated to any observables, as all neighbors independent of direction and vision shadowing are taken into account. This is where the task becomes more complex for an individual agent to find its optimal position within the dynamic group. The negative reward (also referred to as “collision penalty”) has a short range $1/r^3$ distance dependence with a fixed factor of $-3$ times the positive reward, which would result in an maximal reward and therefore optimal inter-particle distance of $3\sigma$ if only considering two agents in the system. Note, that rewards are not diverging for small distances, as they are limited by the minimal particle distance of $\sigma$ due to the physical size of the experimental particles.

7.2 Training process

Indeed, after training the policy under this reward, the active particles perform a swirling motion as a cohesive, but well-spaced group which is rotating globally in a counter-clockwise direction. The experimental snapshot shown in Fig. 7.3 (A) displays 30 active particles which have been experimentally trained for about 43 h at the time of recording and are annotated with their trajectories of the past 10 min to visualize their motion. The training process leading to this behavior is represented by the time evolution of relevant quantities displayed in Fig. 7.3 (B). While the definition of the reward represents a infinite task (i.e. there is no final target to reach) the training has been divided into experimental measurements of 1 h each, after which all particles in the field of view were pushed back to a central initial position before training has been continued. This mainly
7.2 Training process

Figure 7.3: (A) Snapshot of the experimental system, comprised of 30 active particles recorded by bright field microscopy after RL training has converged. Particles are annotated with their trajectory of the last 10 min. (B) Time evolution of relevant parameters during training of the experimental system. Data points are averaged over all particles and per individual measurement, taking 1 h each. The dashed line indicates measurement number 43, in which the snapshot shown in (A) has been recorded.

served the purpose of avoiding the whole group to drift into the measurement boundaries, where the quasi-reflective reorientation protocol would need to handle many particles and would degrade the trajectories recorded to train the policy. Furthermore, it might have helped as a hint in early training, to start each measurement in a state of a relatively compact group with corresponding high reward. Each experimental measurement itself has been divided into three batches of 20 min (corresponding to 120 actions performed) after each of which all recorded trajectories have been evaluated and bootstrapped to train both neural networks.

The progress of training is best represented by the growth of mean reward of the agents over time, which converges to a maximum while the actor is converging to the optimal policy. Another good indicator is the mean Shannon entropy

\[
H(a) = - \sum_{i=1}^{3} p(a_i) \log_3 (p(a_i))
\]  

(7.5)

of the policy output, which is 1 if all actions are equally likely to be chosen and 0 if one action is strictly preferred over the other two. Naturally, the entropy is expected to drop during the training process, however, it might converge to a finite value as especially for more complex tasks, more than one option might be similar advantageous, depending on the situation.

For the task of swirling motion (or more precise, aggregation with collision avoidance) there are two regimes which can be observed in the time evolution given in Fig. 7.3 (B): The group density, which is measured as the sum of all inverse particle distances quickly increases and the optimal
density is already reached after just 15 h. This is much faster than the increase in local particle alignment and the correlated global rotation $O_R$ of the group performing the swirling motion. The reason for this is likely the fact that group density and the corresponding observable $p_d^m$ are highly correlated to the reward for initial aggregation, which can also be seen in the rapid increase in mean reward in the first 15 h. After this point, the mean reward continues to increase, but much slower, while the RL framework is basically “fine-tuning” the policy, to stay at an optimal density while also keeping active particles evenly spaced to avoid any penalty for collisions. As expected, this is achieved by increasing the local alignment of particles, which is correlated to the observables $p_{m}^{\sin}$ and $p_{m}^{\cos}$ but not directly correlated to the reward function.

The fact that the reward (and all other shown quantities) are declining towards the end of the series of experiments shown in Fig. 7.3 (B) is not due to instabilities in the training algorithm, but due to degradation in the experimental sample. This includes mainly active particles loosing their carbon caps after multiple days of constant propulsion, or active particles becoming stuck at the substrate or to each other. As those particles do not properly react to the applied laser patterns any more, particle actions become ill-defined and the training experience sourced from their trajectories misguides the RL framework and can degrade the policy. To exclude such effects from experimental training, only measurements taken before any sample degradation visual in the video recordings are taken into account for evaluation. While in the case discussed here the network was assumed to have converged to a sufficiently good policy, further training could have been performed with a new set of previously unused particles.

### 7.3 Final configuration

As noted above, the particles form a well-spaced cohesive group which performs swirling motion after the policy has converged. The distances between particles are visualized in Fig. 7.4 (A), where the same snapshot as in Fig. 7.3 (A) is shown again, however, this time annotated with connections between all particles instead of trajectories. The green lines represent all particle-particle connections, which distances are the sole source of reward. The blue lines on the other hand displays the network of next neighbors. While direct neighbors do not receive any special treatment in respect to the reward function, the distance to its next neighbors in front is what an individual agent can control best. Here, next neighbors have been defined by a Delaunay triangulation with post-applied filtering for edges opposite of obtuse angles. The probability distribution of these particle distances is shown in Fig. 7.4 (B). While it is clear from the main peak of the blue histogram (next neighbors) that particles indeed try to keep some optimal distance between them, it is also
7.3 Final configuration

Figure 7.4: (A) Snapshot of the experimental system, showing the same configuration as in the previous figure, but annotated with next neighbor distances (blue) and distances between all particles (green), which are the base of reward calculation. Next neighbor distances are defined by triangulation with a post-applied filter to remove any connections opposite of obtuse angles. (B) Histograms of next-neighbor and all-particle distances (left axis) and the pairwise reward function (black line, right axis). Note that the reward drops significantly to a value of $-2$ for the minimal particle distance of $\sigma$, which is not visible in the figure above.

It is obvious that collisions still frequently occur, as the probability is still high and even shows a small peak at exactly particle diameter $\sigma$. This is, of course, due to the dynamic and noisy nature of the system and the environment. For the same reason, the mean distance to next neighbors is larger than what would be expected for a noiseless passive system with a similar “potential”. While the total reward could be higher if particles were perfectly even spaced with a shorter distance, possible and likely fluctuations in the particle distances result in much worse reward when particles come closer to each other, than if they drift a bit further away from each other. Hence, in order to optimize the reward, it is of advantage to increase the mean particle distance to account for the unsymmetrical negative impact of fluctuations.

For further inspection of the trained behavior of the particles, it is helpful to directly inspect the output of the policy, i.e., the typical actions chosen for a given situation. This is done in Fig. 7.5, where the distribution of all chosen actions is plotted in bins of the corresponding state of particles. Figure 7.5 (A) shows the average actions taken by a north-facing particle in respect to its relative position to the center of mass of the particle group. The most notable feature is the strong tendency to do a left turn, whenever the reference particle is in front of the group center (red region in the top half of the plot). While the situation is less clear if the particle is facing the group center straight ahead or left to the left, there is a strong tendency to do a right turning motion if the group center is located towards the front right of an agent (green region in the lower left of the plot). The
7 Swirls of Active Particles based on Reinforcement Learning

Figure 7.5: Visualization of the RL policy after training. (A) Probability distribution of actions chosen by a north-facing particle (inset) depending on its position relative to the center of mass of the particle group. (B) Probability distribution of actions of a north-facing particle depending on the mean orientation of observed neighbors relative to itself (indicated by purple arrows). For both plots, the chosen actions of experimental trajectories of one hour of measurement have been sampled and binned by the respective quantities. Bins are color coded by the ratio of sampled actions, which can be interpreted as barycentric coordinates into a triangular color plane with blue indicating forward motion, red indicating a left turn and green indicating a right turn (triangle shown on the right hand side). The gray contour lines mark the amount of sampled actions per bin; bins with insufficient statistic are replaced with a gray-checkered background pattern.

The gray contour lines plotted on top of the colored bins represent the number of data points sampled for each bin and indicates that there is a typical (and likely optimal) position or local state where particles are located most of the time. This optimal position is tangentially aligned on the right side of the group center, as one would expect for a counter clockwise rotating swirl and spans out from the center to about 50 µm, which is the typical radius of the group as seen in Fig. 7.3 (A).

Note that, in contrast to the experiments shown in Chapter 6, a spontaneous reversal of the direction of rotation of the group is principally impossible with this policy: In the example shown above, particles have learned to collectively rotate in a strictly counter-clockwise swirl. However, as the reward function is symmetrical in this regard there is no advantage between rotating clockwise or counter clockwise and there should be an equal optimal policy which performs a clockwise swirl. Importantly, these two policies form local maxima in the hyperplane of the policy performance function $J(\theta)$, introduced in Eq. (3.8) as the expected return of the initial state. Hence, in contrast to Chapter 6 where the interaction rule is symmetric and symmetry breaking appears instantaneously during application of the rule, here, symmetry breaking is part of the training process and happens during gradient ascend of the policy parameters.
7.3 Final configuration

Looking at the distribution of commonly chosen actions, it also becomes apparent that particles rarely chose to do a plain “forward” motion. This has two likely reasons: As discussed above, the main mechanism for collision avoidance is the development of local alignment. Consequently, particles will always try to optimize their local alignment by adjusting their orientations to their neighbors, while at the same time they do not have any localized target to move forward to. Additionally, most of the particle motion is governed by left turning actions to follow the collective swirl. As the particle speed in turning actions is slightly lower than in forward actions (about two thirds), this lower speed predominates the average particle speed in the swirl. In order to keep a good distance within the swirl, it might be of advantage for individual particles to “wiggle” left and right even if they are well aligned, to have an effectively slower forward motion. That this is in fact the case, can be seen well in Fig. 7.5 (B), where the chosen actions are plotted directly over the observables for orientation of neighboring particles. The y-axis of the plot shows the longitudinal (i.e. in swimming direction) alignment of neighboring particles with the agent, which is given by the sum of the corresponding observables $\sum_m p_m^{\cos}$. The x-axis shows the side-wards component of mean neighbor orientation, in this thesis referred to as lateral alignment of neighboring particles, given by $\sum_m p_m^{\sin}$, respectively. If particles are well aligned with their neighbors, i.e. positive longitudinal alignment, left and right turning actions become similar likely as a direct result of the given observables but forward actions are not favored by the policy (yellow/orange area at the top center of Fig. 7.5 (B)).

Another noticeable feature of Fig. 7.5 (B), is the strong preference for left turns if the mean orientation of neighboring particles is close to zero in both components. While this combination of observables can be caused by randomly oriented neighbors, the statistics here will be dominated by situations where no neighboring particles are visible to the agent at all. If the agent has no other group members within sight, for an efficient policy, it should turn around until it sees some parts of the group again to orient itself. However, this process might take several turning actions, which should persistently rotate the particle in one direction. As the policy does not include any short time memory of previously chosen actions, this persistence has to be encoded in the policy. This means that the policy has to converge to one direction of rotation which is strictly chosen over the over in this situation. So, again, symmetry breaking into either direction is part of the training process. At the same time, an agent without sight of any neighboring particles is likely facing outwards from the group center near the edge of the group. Consequently, it should rotate itself in the same direction as the swirl, to efficiently realign itself. Thus, the local reorientation of particles and the global direction of rotation of the swirl are highly correlated and both default to a left turn or counter-clockwise rotation in case of the presented instance of a trained policy.
7 Swirls of Active Particles based on Reinforcement Learning

7.4 Discussion

While trying to optimize the reward of keeping agents in general proximity but avoiding direct contact, the RL framework has converged to a policy which causes the active particles to perform a stable swirling motion. Opposed to the swirls in Chapter 6 which spontaneously can change their direction of rotation, here the optimal solution to this task includes a broken symmetry in the policy, which results in the particle group always rotating in the same direction. One reason for this early symmetry breaking in the training process is given above: The lack of short term memory in situations where an agent has no neighbors within sight. Yet, this is only relevant due to the choice of a finite vision cone in the definition of our observables in the first place. If an agent would have full $360^\circ$ vision, it would always be able to estimate the position of the group center, rendering the need for any memory or fixed direction for a persistent rotation obsolete. In fact, this lack of more global information available to the decision making process is one of the factors which also creates the necessity for a swirl: Assuming that collision avoidance is satisfied best by perfect alignment of neighboring particles, the maximum reward could also be achieved by the agents when forming a polarized flock. However, to keep organized as a flock, each individual has to know its position relative to its neighbors, including the agents at the front of a flock — which is not possible if only forward vision is available. These effects of finite vision angles on collective behavior are well known and have been considered as an important factor for example in modeling schools of fish (Couzin et al. 2002; Gautrais et al. 2008). In fact, auxiliary simulations performed alongside the experiments discussed in this chapter have shown a chain-like behavior of particles following each other when only visible neighbors have been included in the reward function, not unlike the behavior found by Barberis and Peruani (2016) in their model of social forces. Only the more general reward given in Eq. (7.4) has lead to the more natural swirls shown above.

A more experimental factor, why a swirl might be additionally preferred over a flock is the heterogeneity in particle motion. Varying speeds between individual particles in the group have less impact if the group as a whole is in a stationary configuration. Slower particles might stay closer to the center of a swirl, while faster particles might be more to the outside but always in proximity to their neighbors. In a flocking motion on the other hand, not only directional alignment, but also velocity matching between neighbors is important to keep up cohesion of the group, rendering a flock less stable and robust against perturbations than a swirl.

When comparing the swirls discussed in this chapter to the swirls discussed in Chapter 6, the two most notable differences are particle density and the point at which the symmetry of the system is broken. The particle density is no conceptual difference and should be tuneable in both systems by
the corresponding radius of repulsive force or the cost of collisions, respectively. The difference in symmetry breaking is more fundamental, however, it is not necessarily an artifact of reinforcement learning, but can be narrowed down to the precise problem definition. The underlying idea of the social interaction rule in Chapter 6 has been to model swirling behavior, but individual particles have been explicitly allowed to adapt to their environment, namely the orientation of their neighbors. The definition of reward in this chapter on the other hand has been specifically to find a collective state in which particle distances are kept at an optimum, which resulted in a policy to perform swirling motion in the most stable and robust way. When revisiting the hysteresis measurements discussed in Section 6.4, we can see that the average rotational order parameter – a good indication for the stability of the swirling motion – has in fact been slightly higher for cases of a strong bias in the direction of rotation. Optimizing the stability of the swirling motion by all means would therefore also include a fixed symmetry breaking in the case of a social interaction rule. This illustrates the importance of such details in the definition of reward, and a more general characteristic of reinforcement learning: Any optimal policy will only be as adaptable to changes in the environment as strictly necessary to solve the given task. In return, if reinforcement learning is employed to study collective behavior, the definition of reward has to reflect to complexity of a problem space in which rich collective behavior can emerge as a viable solution.
Collective Behavior in Presence of Randomly Appearing Food Sources

Creating a complex task to be solved by the reinforcement learning algorithm does not necessarily require a complex definition of rewards. In fact, the experimental environment of our active particles is already rather complex and a definition of reward can be simple, but still able to trigger complex particle behavior if it requires non-trivial navigation through the surrounding environment. To design a simple task with these features we took inspiration from natural systems and added a (virtual) food source to our environment. It is implemented as a circular region of fixed diameter, located at a random position within the experimental area. Individual agents gain a constant reward as long as they are inside this area and no reward otherwise. Importantly, no rewards or penalties are given with respect to any of the other particles in the system. Thus, the reward has no direct incentives for any collective behavior, which makes the task at hand a-priori a selfish problem. Additional complexity is added to the system by modeling consumption of food source, making it non-stationary. The food source is initialized with a finite capacity which is reduced by every reward given to an agent. If the capacity is exhausted, the food source disappears and is reinitialized at a new random position somewhere else within the measurement area. Consequently, it is not only important for the agents to reach and then stay inside the food source, but also to be able to regularly migrate to the new location of the food source, once it is depleted at the current location.

The implementation of reinforcement learning is the same as described in the previous chapter, with 30 particles sharing a central policy. One important difference is the addition of a fourth set of observables for food perception,

$$f_{itm} = \frac{\sigma_{\text{food}}}{|r_i - r_{\text{food}}|} g_{\text{food}}_{itm}. \quad (8.1)$$

leading to a total of 20 observables with $p^d_m, p^\text{sin}_m$ and $p^d_m$ defined as in the previous chapter (Eqs. (7.1) to (7.3)). Here, $\sigma_{\text{food}} = 80 \mu m$ is the food diameter and $g_{\text{food}}_{itm}$ represents the fraction of food visible
8 Collective Behavior in Presence of Randomly Appearing Food Sources

Figure 8.1: Conceptual representation of observables. Each agent is assumed to have a 180° forward vision divided in five sections $m = 1 \ldots 5$, for which observables for neighboring particles (lime green), their orientation (purple) and, contrary to Fig. 7.2, a virtual food source (orange) are calculated. The gray areas correspond again to the blind angle and shadowed regions for which the reference particle has no vision.

...to the $i$th particle in section $m$ of the vision cone at time $t$. Analog to neighboring particles, the food source is also subject to vision blocking, as indicated by the gray shadows in Fig. 8.1 (A). Note that the normalization of the observables is not strictly necessary, as the weights of the first layer of the neural network allows for arbitrary scaling, however, it is of advantage for the gradient ascent if observables are equally scaled and therefore improves convergence of the network. The difference in number of observables of course also changes the input dimensions of the two neural networks, but all other layers have not been altered in comparison to the previous chapter. The formal definition of reward is simply given by

$$r_i^t = \begin{cases} +1 & (|r_i^t - r_{\text{food}}| \leq \sigma_{\text{food}}) \\ 0 & \text{otherwise} \end{cases} \quad \text{(8.2)}$$

8.1 Emergent collective behavior

Similar to the previous chapter, Fig. 8.2 (A) shows a snapshot of the experiment after the policy has converged. The trajectories of individual particles are plotted on top of the microscope image, alongside with the current food source (orange circle) and the location of the previous food source (dashed circle). Unsurprisingly, particles have learned to swim towards the new food source in a relatively straight fashion, after the previous one has been exhausted. As a direct result of all particles swimming towards the same direction, the group forms a polarized flock during this migration. More notably, when the group has reached the new food source, a swirling motion is established. This can be seen both by the circling trajectories in Fig. 8.2 (B), as well as the rise of the rotational order parameter $O_R$ which is defined in the same way as in previous chapters. Rotational order is strongly correlated to the fraction of particles which are already inside the food
8.1 Emergent collective behavior

Figure 8.2: Collective behavior with the final policy. (A) Snapshot of the experimental system of 30 active particles after the RL training has converged. Particles are recorded by bright field microscopy and are annotated with their trajectories of the last 40 actions. The solid orange circle shows the current location of the (virtual) food source, the dashed orange circle shows the previous location of the food source before it has been exhausted. Particles move towards the new food location in a relative straight fashion. (B) A later instance in time. After particles have reached the food source, they start to move in a counter-clockwise swirl inside it, as can be seen from the circular trajectories. (C) Quantitative analysis of the swirling motion, given as the rotational order parameter $O_R$ over time (blue line), which is strongly correlated to the amount of active particles inside the area of the food source (orange line).

source and grows to around 0.5 if all particles have reached the food source (Fig. 8.2 (C)). While this is considerably smaller compared to the swirls discussed in previous chapters, the observation is persistent over time and food sources. This illustrates, that despite the selfish nature of the definition of reward, collective behavior still emerges in the group.

To understand why particles form collective behavior, we have to account for the potential ways in which individual particles can interact with each other. These are physical interactions caused by the extended particle bodies and the surrounding fluid as well as interactions in the “information space”, as the particles shape also block the visual input of other other agents, thereby limiting their information on e.g. the food source. The latter is especially important during migration, where individual agents are effectively rewarded for getting to the new food source as fast as possible. Any agent at the front of the migrating flock should be able to perfectly infer the location of the food source from the available observables and can head straight towards it. However, the further back an agent is located in the flock, the more its vision is blocked by preceding particles and its estimate about the exact position of the food source will get significantly more uncertain. On average, about 10% of particles do not see the food source at all (i.e. $\sum_m f_m = 0$) during migration as a result of having their vision completely blocked by neighbors. Yet, even those particles head into the correct direction most of the time, which illustrates that they have learned to orient themselves at and align with their neighbors instead.
The fact that agents have learned to "rely" on their neighbors as a good strategy is also represented in the estimated value for corresponding states. Figure 8.3 shows the output of the critic neural network, representing the estimated value of the current state of a particle as color-coded bins over the sum of food observables $f_m$ and sum of neighbor observables $p_m^d$. It is trivial that the estimated value is higher if the food observables are high, as this indicates that the agent is close to the food source and will soon be rewarded. Interestingly, if the amount of observed neighbors is high, the estimated value of the local state is comparable high even for low amounts of observed food. This indicates that the RL model has indeed learned that the direct observation of food is hindered by having many neighbors, but the real distance to the food source and, thus, more reward, might be closer than what is implied by the food observables alone. This anti-correlation between food and neighbor observables in respect to the value is highlighted by the dashed line in Fig. 8.3. Another feature seen in the plot is that proximity to food is valued even higher, if there are very few to no neighbors visible at all (i.e. $p_m^d = 0$ at the lower edge of the figure). This represents the fact that if an agent is swimming towards the food source but does not see any neighbors, it has to be at the leading edge of the migrating flock. Consequently, it will reach the food source earlier and will have more time than the average group member to collect reward, before the finite capacity of the food source is exhausted. This is reflected by a higher expected return for the corresponding state which is the definition of the estimated value.

Once the group has arrived at the new food location, formation of swirling motion is actually a natural choice. While there is no explicit penalty for contact with neighboring particles in the reward function, there are still good reasons for the group to favor an ordered state while being inside the food source. The main factor which is driving the behavior of individual particles is the ability to collect reward. In order to do so, it is of advantage to individual particles to be able to react to changes in their environment as good as possible. Consequently, every state which limits the motion of a particle should rather be avoided. This includes collisions with other particles,
8.1 Emergent collective behavior

Figure 8.4: Behavior of active particles in simulations after training has converged. (A) Snapshot of a simulation with the same parameters and rewards which have been used in the experiments. The image shows rendered particles (gray spheres) with the trajectories of their last 40 actions inside the food source (orange circle). Contrary to the experiments, particles do not perform swirling motion, but form an immobile jammed cluster (indicated by light gray arrows). (B) Snapshot of a simulation with the same parameters, but a contact penalty added to the reward definition. As indicated by the trajectories and the gray arrow, global rotation of the group similar to the experimental results can be observed.

As these already by geometrical considerations lower the ability of a particle to freely move in its environment. In the case of active particles in real experiments, additional physical interactions reduce the ability of an agent to react to the environment even further. Here, these include possible hydrodynamic as well as phoretic forces that can occur between active particles in close contact. The decline in possible particle motions is also enhanced if multiple particles collide into some form of jammed cluster which can not easily be resolved by the involved agents, but can drift out of the food source or hinder a fast migration once the current food source is depleted.

While all these physical interactions between agents are hard to quantify on their own, they are certainly present in the experimental system and seem to be detrimental enough to the steering capacity of individual particles that the policy has learned to avoid direct contact. This becomes especially clear when comparing the experimental observations with results obtained from simulations: Figure 8.4 (A) shows a snapshot of a simulated system with the same parameters and reward, where hard body interactions are modeled with a Weeks-Chandler-Andersen potential and particles move according to Langevin dynamics. Interestingly enough, particles do not form any swirling motion, but just stay inside the food source in a dense jammed particle cluster. Only if we add a contact penalty to the reward definition\(^1\) swirling motion can also be observed in simulations, as shown in Fig. 8.4 (B). Note however, that even in this case the observed rotational order \(O_R\) in the simulation reaches only about half of what is obtained from experimental results shown

\(^1\) Formally, this contact penalty is defined by adding a penalty of -0.5 to the reward of any particle, which has a center-to-center distance of less than 1.2\(\sigma\) to any other particle in the system.
Figure 8.5: Mean reward gained by particles in simulations where some observables have been zeroed out and training has been performed with only food observables \( \{f_m\} \), only a combination of food and neighbor density observables \( \{f_m, p^d_m\} \) or all observables \( \{f_m, p^d_m, p_m^{\sin}, p_m^{\cos}\} \) respectively. The mean reward is increasing with the number of available observables for both simulations with the original reward definition (red line), as well as simulations with the altered reward definition including a contact penalty (blue line). Data is shown for the best out of 15 trained policies per data point. The error bars correspond to the variance in mean reward within 5 evaluation episodes for each chosen policy.

in Fig. 8.2 (C). This illustrates, that even with the added contact penalty, the simulation is not able to capture the full complexity of the experimental system.

Another aspect highlighted by the simulation results is the importance of the orientation observables \( p_m^{\cos} \) and \( p_m^{\sin} \) to the final policy. Figure 8.5 displays the mean reward obtained in simulations where some observables have been zeroed out before passing them to the neural networks. Intuitively, the mean reward will be lower if only the food observables are present to an agent. As vision can still be blocked by other particles, the reduced food observables during migration will affect the overall performance of the policy. Notably however, even in the simulations without a contact penalty (red line), which do not show any swirling behavior, the overall performance is increased if information about the orientation of neighboring particles is available. This means that even in this case, alignment with other particles is still a non negligible aspect of the final policy.

While alignment with neighboring particles is an important aspect, the final policy obtained from experimental training is also strongly governed by forward motion (Figs. 8.6 and 8.7). This is a considerable difference to the policy of the previous chapter, but can be easily understood as the major aspect to optimize policy performance is a fast migration towards the new food source in case the current one is depleted. Once the group has reached the next food source, it is of course also important for all agents to stay in side its limits. As being part of the then swirling group coincides with being inside the food source, there is again a sweet spot which each agent is trying to reach in relative to the center of the group. The distance of this position to the center of mass of the group is about 20 \( \mu m \), which corresponds to half of the radius of the food source and results in the formation of a swirl which is small enough to fit within it. Contrary to the previous chapter, it is again important for agents to reach this position relatively fast, which results in a clear separation
8.1 Emergent collective behavior

**Figure 8.6**: Visualization of the policy after training has converged. (A) Probability distribution of chosen actions sampled from the final measurement for a north-facing agent depending on its position relative to the center of mass of the particle group. The gray contour lines mark the amount of sampled actions per bin; bins with insufficient statistic are marked by the gray-checkered background pattern. (B) Probability distribution of chosen actions for artificial states of a solitary north-facing particle depending on its position relative to the food source. These states correspond to the situation of an agent leading the migrating flock, for which all other particles are located in its blind area at the back side.

of necessary actions that an agent has to perform. This separation is highlighted by the dashed lines in the map shown in Fig. 8.6 (A).

When looking at the actions of a single particle with respect to the position of the food source, it becomes once more apparent how experience about the system is comprised in the policy. The actions shown in Fig. 8.6 (B) correspond to the policy output of a solitary north-facing particle which only observes the food source but no neighboring particles. This is representative for the agents at the leading edge of the flock when migrating from an exhausted food source to the new one. Obviously, the agent is trying to swim towards the food source and rotating to it if located to either side. However, it is not actually aiming directly for the center of the food source but rather aims to enter on the right hand side. Broadly speaking, even the first agent of the group to arrive at the food source is “aware” that a swirling motion will develop inside it and already starts to align with it, even though it is not yet present.

The preference to enter the food source at an slight angle on the right hand side can also be seen when mapping the chosen actions directly over the locally perceived observables. Figure 8.7 shows the chosen actions sampled from particle trajectories of the final experimental run mapped over the food observables \( f_m \) (A), the particle density observables \( p_d^m \) (B) as well as the the orientational observables \( p_m^{\sin} \) and \( p_m^{\cos} \) (C). The former two are mapped over their total sum as well as the mean angle of their distribution. For food observables, it is clearly visible that agents swim towards the
8 Collective Behavior in Presence of Randomly Appearing Food Sources

Figure 8.7: Visualization of the final policy in direct dependence of different sets of observables. All actions are sampled from the final experimental measurement and are mapped over (A) the total amount \( \sum_m f_m \) and average direction of observed food, (B) the total amount \( \sum_m p_m^d \) and average direction of observed neighboring particles, as well as (C) the lateral \( \left( \sum_m p_m^{\text{lat}} \right) \) and longitudinal \( \left( \sum_m p_m^{\text{long}} \right) \) alignment of neighboring particles. Average directions are calculated by the first angular moment \( \frac{\sum_m \phi_m f_m}{\sum_m f_m} \), where \( \phi_m \) denotes the center directions of the sections \( m = 1 \ldots 5 \) of the vision cone. The individual sections are indicated in the plot by dashed lines.

food source in a more or less symmetric fashion for medium to large distances (i.e. \( \sum_m f_m < 0.6 \)). If an agent is close to the food source however, it will only continue to move forward if the majority of observed food is located to its left (positive angular direction), while a right turn is favored if most of the observed food is located to its right hand side (negative angular direction).

The dependence of observed neighboring particles to the chose action is more complex and the response to the respective observables is less clear (Fig. 8.7 (B,C)). Nevertheless a trend to “follow the group” is visible still visible, which shares the same asymmetry of moving preferably forward if the center of observed neighboring particles is slightly to the left of the agent at around 18°. Similar to the policy of the previous chapter a strict left turn is preferred if no other particle is observed to rotate and find the group again as efficient as possible. Note however, that this is not the case if a food source is observed at the same time, as it is the main source of reward and consequently more important than any collective behavior.

8.2 Policy extrapolation

A major advantage of performing reinforcement learning in an actual experiment is the fact that agents learn to handle the more complex and noisy environment. This results in a policy which is likely more robust to disturbance of any kind and should be able to handle unexpected changes to
8.2 Policy extrapolation

Figure 8.8: Behavior of active particles in the absence of food. (A) Snapshot of an experimental evaluation in which particles act according to the final policy, but the food source has been removed from the environment completely. (B) Rotational order in the absence of food. The food source is present for the first 100 actions and removed afterwards. The plot shows the time evolution of the rotational order parameter $O_R$ for 10 iterations (light blue lines) and their average (dark blue line). Notably, the strength of the rotational order does not change significantly if the food source is not present any more.

the environment better than a policy trained in more idealistic simulated scenarios. This fact becomes especially emphasized when we completely remove the food source from the environment, which is the only source of reward for the agents and thereby the only objective of the original selfish foraging task. The resulting behavior of active particles can be seen in Fig. 8.8 (A): Particles perform a less dense but still cohesive swirling motion. In fact, when we perform experiments with the final policy where the food source is present for the first 100 actions (about 15 min) and then removed from the environment, the recorded rotational order $O_R$ is on average even slightly higher after the food source is not present anymore (Fig. 8.8 (B)).

The structure of the particle group can be explained by two features of the policy: The counter clockwise direction of rotation is definitely based on the general tendency of the group to form a swirling motion in this direction. However, in the original system this swirl is usually formed within the food source, which is not present anymore, which significantly changes the local state of the participating agents. In fact agents do what they learned to do if they do not see any food source: The follow and align with neighboring particles in the system. This stabilizes the swirling motion but also explains the more ring-like structure which is formed by agents following each other without an incentive to keep the group dense and compact.

It is important to note that the observed behavior is by no guarantee the optimal behavior in this scenario, as the global state of the system without the food source has never occurred in the training process. Thus, the policy is extrapolated to a region in the global space state, for which it has never been optimized. The fact that agents perform a stable motion is due to the fact that their local state, represented by the observables, matches a situation which also occurred during the
training process. As a result, the agents in the group follow and align with neighboring particles in expectation that there should be a food source in the system somewhere.

Yet, performing swirling motion in the absence of food might actually be the best strategy if the food source is at least expected to reoccur somewhere again. The area in which the food source has to occur is limited by the experimental measurement area, but the agents are not aware of any boundary to the system. So in order to stay close to where the food has to reappear in lack of any landmark in the system the best behavior for individual agents should be to stay approximately at their current position while also not getting trapped in collisions with other particles. Hence, the optimal collective state of the group can be expected to be an ordered but stationary configuration, i.e. a swirl.

8.3 Discussion

In this chapter, we have modeled a foraging task inspired by natural systems. Despite the selfish nature of the task of active particles only getting rewarded for being inside the food source, we could observe emergent collective behavior. Particles did not only align into a polarized flock to efficiently reach the next food source when the previous source has been depleted, but also kept alignment inside the food source to form a swirling motion filling the available area of reward. In contrast to the previous chapter, where collective behavior has been incentivized by the definition of reward, here, collective behavior is purely a result of the complex particle interactions present in the system.

The first aspect of particle interactions which is important for the resulting collective behavior is the degradation on steering performance and general jamming on direct contact with other particles. While this is hard to quantify, it is seemingly influential enough to drive general collision avoidance between particles, which is a major contribution to the formation of ordered states like flocks and swirls. We managed to replicate the experimentally observed alignment to some degree, by adding an explicit cost of collisions to the reward function in our simulations, yet, the swirling motion observed in the experiments is still considerably stronger than what has been achieved in these simulations, underlining the importance of performing real world experiments in the context of reinforcement learning. This discrepancy between simulations and experiments is commonly referred to as reality gap and well known in literature (see e.g. Jakobi et al. 1995; Salvato et al. 2021).
Another component of important particle interactions is the obstruction of view of particles to each other. Importantly, this leads to the need of a “fallback strategy” where agents have to decide what to do in a case where they do not see the food source. If no information about neighboring particles would be available, the only option left would be to just turn around and look out until the food source is visible again. This is precisely what agents have learned to do in simulations where all other observables have been zeroed out. The more efficient strategy as seen above, however, is to follow neighboring particles in the direction they are swimming, relying on the “collective vision” even without seeing the food source as an individual agent. Notably, this fallback behavior of just following the group in case of doubt, is what also enables stable collective motion for an individual agent in the unexpected and untrained scenario of completely removing the food source from the system. While the behavior of agents in this scenario is an extrapolation of the trained policy and can not be expected to be optimal a priori, the observed swirling motion is likely not the worst possible behavior. In fact, given the arguments in the previous section for agents to stay at the current position while possibly avoiding collisions with each other, a swirling motion might actually be the best thing to do for agents in this specific scenario. Another possible advantage of a swirling motion not discussed so far is the increased collective vision of participating agents. A swirl with tangentially aligned particles should maximize the number of particles facing at least partially outwards without obstruction while keeping the group cohesive at the same time. This might be of advantage to spot and collectively reorient towards a new food source both in the case of depletion of the current food source, as well as being in a situation where no food source is available at all. It also marks another example, how individual agents can improve from collective behavior even without being rewarded for doing so directly, which is likely the case in most collective systems.
In this thesis, I present an experimental model system of light-activated artificial microswimmers which can be steered by adjusting propulsion speed and active torque in a feedback loop. We use this system to investigate different approaches and scenarios in modeling collective behavior.

In the first experiments presented in this work, we have employ a social interaction rule, for which we see a critical transition in the collective dynamics from an unordered but cohesive swarm to a rotationally ordered swirl depending on a single angular parameter. We are able to describe this transition well with a simple coarse-grained model, derived solely from symmetry arguments. Importantly, we are able to measure and describe key properties of criticality, like a distinct bifurcation point and critical slowing down in an out-of-equilibrium system of relatively limited size. We are also able to bias the preferred direction of rotation to provide a direct measurement of susceptibility and a distinct hysteresis loop above the critical angle. Our analysis is not only of interest when discussing robustness of such interaction rules in complex and noisy real world systems, but might also provide further hints to the question of criticality in collective behavior. Similar to our system, natural groups are often highly dynamic and equally limited in size, which makes common field-theoretical descriptions difficult to apply when discussing the observed phenomena.

Following the social force based interaction rules, we also apply the framework of multi-agent reinforcement learning, to train a behavioral policy according to which agents interact with each other and their environment. Inspired by the concept of social forces, we first employ a reward function shaped as a “social potential”, with an optimal inter-particle distance to be achieved by the trained policy. As expected, we could observe the emergence of swirling motion with the policy converging towards optimal reward. In contrast to the previous experiments, however, rotational symmetry is not spontaneously broken during motion, but early in the training process, rendering the final swirl more robust to perturbations by only allowing one direction of rotation.
9 Conclusions

To create a more complex task, in order to enforce more comprehensive collective behavior, we subsequently simplify the reward definition. Inspired by natural foraging, we add a virtual food source to the environment, on which agents have to feed. Despite the notion of selfishness in this task, collective behavior emerges in the group of active particles, which can mostly be attributed to the complex interactions of individual agents. These consist of physical interactions of the microswimmers due to hydrodynamic and phoretic forces, but also due to blocking of sight in the visual input of agents. Interestingly, here, collective behavior, especially the observed swirls inside the food source, emerge as a direct response to the complex and noisy environment of the microswimmers, increasing the robustness of individual agents by using other group members for additional reference. Using neighbors for orientation in the environment is also the reason for extended collective motion and a stable cohesive group even in the untrained scenario of a complete absence of food.

The framework of reinforcement learning proves to be a valuable tool for studying collective behavior. Through the versatility of artificial neural networks, reinforcement learning can model complex processes in rich environments that would be difficult to study with predefined interaction rules. Instead of exploring the mechanisms underlying interactions, reinforcement learning moves the focus on the local information represented by the observables. The learning approach illuminates how much information is necessary in order to perform well in a given task, but also how additional information can influence the resulting behavior. The dependence on information availability has become especially clear in the last chapter, where agents have learned to exploit the perception of other group members in order to optimize their individual reward by acting collectively. Consequently, our observations illustrate that collective behavior can indeed arise solely out of selfish motivation.

At the same time, we have shown that depending on the given scenario, policies determined by reinforcement learning can produce very similar behavior to what can be achieved by predefined interaction rules. Further research should also focus on the question how policies resulting from more complex reinforcement learning tasks can be reduced to their primary aspects in order to express them in the form of simplified interaction rules. This reductionist approach might be especially important when designing artificial collective systems like small robotic applications, where energy efficiency and limited computational power need to be considered.

Independent of the origin of interaction, our experiments highlight the advantage of collective behavior in the presence of noise and natural variability. Acting as a group can robustly balance perturbations and, thus, increase the performance of each individual. This benefit of collective behavior is not only reflected by the many instances of coordinated motion observed in animal
groups, but is similarly discussed on the scale of ecosystem stability (Dalziel et al. 2021) as well as applications in robotic swarms (Brambilla et al. 2013; Rubenstein et al. 2014). Alongside observations in natural systems and computer simulations, experimental model systems that provide the complexity and noise of real environments but the ability of individual control will consequently remain an important tool for future studies of the mechanisms and benefits of collective behavior.
References


References


References


References


References


References


References


