

**UNIVERSIDADE FEDERAL DE MINAS GERAIS**  
**Instituto de Ciências Exatas**  
**Programa de Pós-Graduação em Ciência da Computação**

Paulo Alfredo Frota Rezek

**A Stochastic Approach to Generate Emergent Behaviors in Robotic Swarms**

Belo Horizonte  
2023

Paulo Alfredo Frota Rezek

**A Stochastic Approach to Generate Emergent Behaviors in Robotic Swarms**

**Final Version**

Dissertation presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Advisor: Luiz Chaimowicz

Belo Horizonte  
2023

2023, Paulo Alfredo Frota Rezeck.  
Todos os direitos reservados

Rezeck, Paulo Alfredo Frota.

R467s      A Stochastic approach to generate emergent behaviors in  
robotic swarms [recurso eletrônico] / Paulo Alfredo Frota  
Rezeck – 2023.

1 recurso online (138 f. il, color.) : pdf.

Orientador: Luiz Chaimowicz.

Tese (Doutorado) - Universidade Federal de Minas  
Gerais, Instituto de Ciências Exatas, Departamento de  
Ciências da Computação.

Referências: f.124-135

1. Computação – Teses. 2. Robótica – Teses. 3. Robôs –  
Sistema de controles – Teses. 4. I. Chaimowicz, Luiz.  
II. Universidade Federal de Minas Gerais, Instituto de Ciências  
Exatas, Departamento de Computação. III. Título.

CDU 519.6\*82.9(043)

Ficha catalográfica elaborada pela bibliotecária Irenquer Vismeg Lucas Cruz  
CRB 6/819 - Universidade Federal de Minas Gerais - ICEx



UNIVERSIDADE FEDERAL DE MINAS GERAIS  
INSTITUTO DE CIÊNCIAS EXATAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

## FOLHA DE APROVAÇÃO

A Stochastic Approach to Generate Emergent Behaviors in Robotic Swarms

**PAULO ALFREDO FROTA REZECK**

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. LUIZ CHAIMOWICZ - Orientador  
Departamento de Ciência da Computação - UFMG

PROF. DOUGLAS GUIMARÃES MACHARET  
Departamento de Ciência da Computação - UFMG

PROF. RENATO MARTINS ASSUNÇÃO  
Departamento de Ciência da Computação - UFMG

PROF. VALDIR GRASSI JÚNIOR  
Departamento de Engenharia Elétrica e de Computação - USP

PROF. DAVID JULIÁN SALDAÑA SANTACRUZ  
Department of Computer Science and Engineering - Lehigh University

Belo Horizonte, 4 de dezembro de 2023.

# Acknowledgments

First and foremost, I extend my deepest appreciation to my beloved wife, Caroline, whose unwavering love, support, and understanding have been a constant source of motivation and strength throughout this journey.

To my entire family — especially my parents, Paulo and Alessandra, and my siblings, Ricardo and Rafaella — I am eternally indebted for their boundless love, unwavering encouragement, and support, which have been the bedrock of my accomplishments. Special gratitude is extended to my parents-in-law for their remarkable friendship and warmth, embracing and supporting me with open arms. Additionally, I dedicate this dissertation to my beloved grandfather, Zezinho Rezeck, a beacon of wisdom, brilliance, and boundless kindness. Although he may no longer be with us, his legacy lives on through the values he instilled in me and the inspiration he continues to provide.

Furthermore, I owe a debt of gratitude to Professor Luiz Chaimowicz, whose expert guidance, mentorship, and friendship have been pivotal in shaping my academic growth and success. His support and provision of essential resources have been indispensable in my pursuit of knowledge and expertise in this field. Sincere appreciation also goes to Professor Renato Assunção, who has played a fundamental role in shaping the ideas presented in this dissertation through his lectures and cooperation.

Finally, I am grateful to my colleagues at VeRLab for their collaborative spirit and shared knowledge. A special acknowledgment is due to the esteemed members of this dissertation committee — Professors Douglas Macharet, Renato Assunção, Valdir Grassi, and David Saldaña — whose suggestions and feedback have significantly enriched the quality of this dissertation. Lastly, I express sincere gratitude to Capes, CNPq and Fapemig for their financial support, playing a crucial role in enabling the realization of this research.

*“Look deep into nature, and then you will understand everything better.”*

(Albert Einstein)

# Resumo

Esta tese apresenta uma nova metodologia que estende os conceitos de Campos Aleatórios de Gibbs (GRFs) para o contexto da robótica de enxame, permitindo projetar mecanismos de controle que produzem diferentes comportamentos de enxame usando apenas informações locais. Nesse contexto, um GRF é um modelo gráfico probabilístico que descreve as interações e comportamentos de um grupo de robôs. Assume-se que os robôs interagem uns com os outros de uma forma que pode ser descrita por um conjunto de variáveis aleatórias. Essas variáveis definem um campo aleatório e a função de probabilidade conjunta é uma distribuição de Gibbs que descreve a probabilidade do enxame estar em uma determinada configuração. Utilizando o método Markov Chain Monte Carlo (MCMC), cada robô amostra comandos de velocidade de forma descentralizada, forçando-os a se moverem em direção ao mínimo global do potencial, o que direciona todo o enxame a convergir para o comportamento desejado. Esta abordagem tem várias vantagens sobre os métodos mais tradicionais para controlar o comportamento de um enxame. Por exemplo, permite o controle descentralizado, onde cada robô toma decisões com base em informações locais, em vez de depender de um controlador central. Isso torna o sistema mais robusto e escalável, pois não existe um único ponto de falha e o enxame pode continuar a operar mesmo se os robôs individuais falharem. Além disso, permite uma abordagem mais flexível e adaptável ao comportamento do enxame, pois a função potencial pode ser modificada para levar em conta mudanças nas condições ambientais ou novos objetivos para o enxame. Para demonstrar a aplicação de nossa metodologia, investigamos o design de métodos que abordam três desafios significativos na robótica de enxame: flocking e segregação, transporte cooperativo de objetos e formação de padrões. Simulações numéricas e experimentos utilizando robôs reais mostram que esses métodos são escaláveis, adaptáveis e robustos, mesmo na presença de ruído, falhas e mudanças

no ambiente. Mais especificamente, o primeiro método mostra ser capaz de segregar adequadamente um grupo de robôs heterogêneos, mantendo a navegação coesa e evitando obstáculos no ambiente. O segundo método demonstra o transporte de objetos de diferentes formas, tamanhos e massas. Também é escalável e resiliente a mudanças na localização do objetivo e falhas nos robôs. Os experimentos do terceiro método mostram a capacidade de criar diversos padrões usando diferentes restrições de vizinhança e que podem servir de base para aplicações mais tangíveis de um enxame de robôs heterogêneos, como a construção de estruturas encadeadas ou semelhantes a pontes dinâmicas. No geral, a metodologia proposta mostra-se promissora e contribui para o campo da robótica de enxame, permitindo a concepção de mecanismos que produzam adequadamente diferentes comportamentos de um enxame de robôs.

**Palavras-chave:** Robótica. Sistema multi-agente. Robótica de enxames. Robótica probabilística.

# Abstract

In this dissertation, we present a novel methodology that extends the concepts of Gibbs Random Fields (GRFs) to the context of swarm robotics, allowing us to design control mechanisms that produce different swarm behaviors using only local information. In this context, a GRF is a probabilistic graph model that describes the interactions and behaviors of a group of robots. The robots are assumed to interact with each other in a way that can be described by a set of random variables. These variables define a random field, and the joint probability function is a Gibbs distribution that describes the probability of the swarm being in a given configuration. By using a Markov Chain Monte Carlo (MCMC) method, each robot sample velocity commands in a decentralized way, forcing them to move toward the global minimum of the potential, leading the entire swarm to converge to the desired behavior. This approach has several advantages over more traditional methods for controlling the behavior of a swarm. For example, it allows for decentralized control, where each robot makes decisions based on local information rather than relying on a central controller. This makes the system more robust and scalable, as there is no single point of failure, and the swarm can continue to operate even if individual robots fail. Additionally, it allows for a more flexible and adaptable approach to swarm behavior, as the potential function can be modified to account for changing environmental conditions or new goals for the swarm. To demonstrate the application of our methodology, we investigate the design of methods that tackle three significant challenges in swarm robotics: flocking and segregation, cooperative object transportation, and pattern formation. Numerical simulations and real-world experiments show that these methods are scalable, adaptable, and robust, even in the presence of noise, failures, and changes in the environment. More specifically, the first method shows to be able to adequately segregate a group of heterogeneous robots while keeping cohesive

navigation and avoiding obstacles in the environment. The second method supports the transportation of objects of different shapes, sizes, and masses. It is also scalable and resilient to changes in goal location and robot failures. The third method experiments show the ability to create diverse patterns using different neighborhood constraints and that it may serve as a basis for more tangible applications such as the construction of chain or bridge-like structures using a swarm of heterogeneous robots. Overall, the proposed methodology shows promise for contributing to the field of swarm robotics, enabling the designing of mechanisms that adequately produce different behaviors of a swarm of robots.

**Keywords:** Robotics. Multi-agent system. Swarm robotics. Probabilistic robotics.

# List of Figures

1.1	Taxonomy of swarm robotics. . . . .	19
1.2	Swarm robotics behaviors. . . . .	22
2.1	An instance of a random field. . . . .	31
2.2	An instance of clique problem. . . . .	32
4.1	Robot state representation. . . . .	40
4.2	Neighborhood system evolving over time. . . . .	43
4.3	The Coulomb-Buckingham potential function. . . . .	49
4.4	Methodology for generating diverse swarm behaviors. . . . .	51
5.1	Flocking-segregative behavior. . . . .	54
5.2	Illustration of segregative flocking. . . . .	61
5.3	Comparison among the minimum number of clusters after segregation. . . . .	63
5.4	Impact of the noise in the sensor model on the performance of our method. . . . .	66
5.5	Flocking-segregative behavior demonstrated in simulated environment $\mathcal{A}$ . . . . .	67
5.6	Flocking-segregative behavior demonstrated in simulated environment $\mathcal{B}$ . . . . .	68
5.7	Flocking-segregative behavior demonstrated in simulated environment $\mathcal{C}$ . . . . .	68
5.8	Flocking-segregative behavior demonstrated in simulated environment $\mathcal{D}$ . . . . .	69
5.9	Flocking-segregative behavior demonstrated in simulated environment $\mathcal{E}$ . . . . .	70
5.10	Mean convergence time to achieve segregation across diverse test environments. . . . .	70
5.11	Method's robustness when faced with robot failure. . . . .	71
5.12	Evolution of the average velocity error among the robots to illustrate the consensus over time. . . . .	72
5.13	Proof-of-concept experiment with one group of ten robots. . . . .	73
5.14	Method's performance evaluation for a single groups . . . . .	74

5.15	Proof-of-concept experiment with two groups of five robots each. . . . .	74
5.16	Method's performance evaluation for two groups. . . . .	74
6.1	Swarm of robots cooperatively transporting an object. . . . .	76
6.2	Transportation strategies using multiple robots. . . . .	77
6.3	Swarm behaviors emerge from the combination of the potential functions. . . .	85
6.4	Perception of the robot regarding the object and its target location. . . . .	87
6.5	Snapshots of four distinct experiments showcasing the system's scalability with varying numbers of robots. . . . .	93
6.6	Scalability of the system when increasing the number of robots. . . . .	94
6.7	Snapshots illustrating the system's adaptability to robot failure and dynamic target location changes. . . . .	95
6.8	Adaptability regarding robot failure occurs and the target location changes. .	96
6.9	Swarm's efforts in the transportation of rectangular prismatic objects. . . . .	97
6.10	Swarm's efforts in the transportation of octagonal prismatic objects. . . . .	97
6.11	Swarm's efforts in the transportation of triangular prismatic objects. . . . .	98
6.12	Robustness regarding different objects with different sizes and masses. . . . .	99
6.13	Simulated robots transporting an object in a complex environment. . . . .	99
6.14	Real robots transporting an object toward its goal. . . . .	100
7.1	Robots mimicking atoms to create emergent chain patterns. . . . .	103
7.2	Exemplification of the Octet rule. . . . .	108
7.3	State of the bond partition for a given robot and its neighborhood. . . . .	110
7.4	Snapshots of four simulated experiments showing different patterns. . . . .	114
7.5	Analysis of the persistence and velocity consensus for different patterns. . . .	115
7.6	Swarm form shapes with a topology similar to a bridge. . . . .	118
7.7	Real robots creating a dynamic bridge pattern. . . . .	118
A.1	Design of the open swarm robotic platform. . . . .	138

# List of Tables

6.1	Main characteristics of research work using pushing-only transport strategy. . .	80
6.2	Average transportation time for when the number of robots increases. . . . .	92
6.3	Average transportation time when adapting to changes in the environment. . .	96

# Contents

<b>1</b>	<b>Introduction</b>	<b>16</b>
1.1	Motivation . . . . .	17
1.2	Problem Overview . . . . .	18
1.3	Objectives . . . . .	21
1.4	Contributions . . . . .	24
1.5	Dissertation Outline . . . . .	26
<b>2</b>	<b>Background</b>	<b>28</b>
2.1	Random Fields and Markov Properties . . . . .	28
2.2	Markov and Gibbs Random Fields . . . . .	30
<b>3</b>	<b>Related Work</b>	<b>35</b>
<b>4</b>	<b>Methodology</b>	<b>39</b>
4.1	Formalization . . . . .	40
4.2	Modeling Robotic Swarms using GRFs . . . . .	42
4.3	Parallel Gibbs Sampling . . . . .	45
4.4	Sampling Algorithm . . . . .	46
4.5	Potential Energy . . . . .	47
4.5.1	Coulomb-Buckingham Potential . . . . .	48
4.5.2	Kinetic Energy . . . . .	49
4.6	Designing Swarm Behaviors . . . . .	50
<b>5</b>	<b>Flocking-Segregative Behavior</b>	<b>53</b>
5.1	Introduction . . . . .	53
5.2	Related Work . . . . .	55

5.3	Method . . . . .	58
5.3.1	Formalization . . . . .	58
5.4	Experiments and Results . . . . .	61
5.4.1	Segregation Analysis . . . . .	62
5.4.2	Flocking Analysis . . . . .	65
5.4.3	Flocking-segregative in Complex Environments . . . . .	66
5.4.4	Robustness Analysis . . . . .	71
5.4.5	Real Robots . . . . .	72
<b>6</b>	<b>Cooperative Transport Behavior</b>	<b>75</b>
6.1	Introduction . . . . .	75
6.2	Related Work . . . . .	77
6.2.1	Transportation Strategies . . . . .	77
6.2.2	Push-based Methods . . . . .	79
6.3	Method . . . . .	83
6.3.1	Formalization . . . . .	83
6.3.2	Swarm Behaviors and Potential Functions . . . . .	84
6.3.3	Cooperative Transport . . . . .	86
6.3.4	Obstacle Avoidance . . . . .	89
6.3.5	Cohesive Motion . . . . .	89
6.3.6	Combination . . . . .	90
6.4	Experiments and Results . . . . .	91
6.4.1	Scalability Analysis . . . . .	92
6.4.2	Adaptability Analysis . . . . .	94
6.4.3	Robustness Analysis . . . . .	96
6.4.4	Object Transportation in Complex Environments . . . . .	98
6.4.5	Real Robots . . . . .	100
<b>7</b>	<b>Pattern Formation Behavior</b>	<b>101</b>
7.1	Introduction . . . . .	101

7.2	Related Work . . . . .	103
7.3	Method . . . . .	107
7.3.1	Formalization . . . . .	107
7.3.2	Potential Energy . . . . .	110
7.4	Experiments and Results . . . . .	111
7.4.1	Diversity of Patterns . . . . .	111
7.4.2	Building Chain-Like Structures with Real Robots . . . . .	116
<b>8</b>	<b>Conclusion and Future Work</b>	<b>119</b>
8.1	Conclusion . . . . .	119
8.2	Future Work . . . . .	122
	<b>Bibliography</b>	<b>124</b>
	<b>Appendix A HeRo 2.0: a Low-Cost Robot for Swarm Robotics Research</b>	<b>136</b>

# Chapter 1

## Introduction

Robots have been at the core of industrial automation for several decades, contributing to the maintenance and evolution of modern society. Specifically, increasing advances in robotics are transforming many industries, from manufacturing to health care, agriculture, transportation, and warehouse management. For the most part, robots were capable of performing tasks with outstanding precision, speed, and consistency, proving to be more efficient than humans in repetitive tasks and even enhanced dexterity. But lately, many have observed that such attributes may not be sufficient for future demand, requiring different robotics paradigms, especially those with multiple robots working cooperatively. For instance, imagine scenarios in which a robot cannot transport an object that is large or heavy or that it has to find some goods in an extremely large search region.

In the last decades, many researchers have been growing interest in developing multiple robot systems that can operate autonomously in a collaborative manner. For many researchers, nature is perhaps the greatest inspiration, providing examples of how successful cooperative systems could be designed. One of the most impressive displays of nature is in the way that particles self-organize to form molecules, how molecules form molecular structures and cells, how cells form tissues and organs, and how organisms arise from and interact with others generating even more complex collective behaviors at each evolutionary step. The examples are numerous, but a fascinating fact is that behaviors like these seem not to be dictated by a central entity but emerge from simple interaction rules inherent to nature. In this context, the concept of emergence describes a process in which the outcome results from the complex interactions of its constituent parts. So the outcome may not be directly obvious and carry a certain degree of unpredictability.

While robotic systems differ from particles or organisms in many ways, the underlying idea of simple rules guiding individual robots leading to collective behaviors coined the concept of swarm robotics within the field of multi-robot systems. As observed in natural systems, swarm robotics takes inspiration from emergent behaviors to design decentralized control methods in a scalable fashion relying only on local sensing and communication capabilities without explicit coordination. Such mechanisms bring additional advantages to robotics, such as better reliability, fault tolerance, flexibility, and scalability. However, generating complex collective behaviors is even more challenging, given that we do not fully understand how complex rules work and also due to technological restrictions, such as limited sensing, communication, and computational capabilities.

## 1.1 Motivation

Despite the potential advantages of swarm robotics, several challenges still need to be overcome before it can reach the operational stage. One of the primary challenges is the limited availability of capable robots that can be produced in a scalable and cost-effective manner using existing technology. Another challenge is the lack of attention and resources given to swarm robotics within the broader robotics community, which tends to focus more on other areas that are being deployed as solutions to current real-world problems. As a result, swarm robotics often receives less funding and support, which can hinder the development and deployment of swarm-based robotic solutions.

Even though researchers in swarm robotics continue to innovate and design novel methods and technologies to address different challenges in robotics, with a focus on scalability and reliability. While many of these developments are currently experimental and limited to simulation and proof-of-concept demonstrations in laboratory or controlled environments, they hold great potential for future real-world applications.

Recently, many researchers have taken advantage of the important aspect that a swarm of robots is composed of simple units, meaning that individual capabilities are

limited. The simplicity of the units holds the potential to reduce their size and produce numerous robots. Even with technological advances enabling the mass production of increasingly smaller and affordable electro-mechanical systems, these swarm robots do not have sophisticated sensing and locomotion capacities restricting the use of typical control approaches. As a result, there has been a growing interest in researching minimalist control strategies for swarms of robots.

The motivation for this dissertation comes from the opportunity to contribute with a novel methodology that can generate a variety of swarm behaviors in a decentralized way requiring only local sensing. In addition to theoretical contributions to swarm robotics, we were also motivated by the affordability of the recent electro-mechanical components allowing improvements in the design and development of a low-cost robotics platform used to verify the method's behavior in addition to simulations.

Moreover, we are highly motivated by the wide range of real-world applications that can benefit from using control methods based on swarm behaviors. Some examples of applications are: search and rescue operations, the contention of oil spills in the ocean, transport of heavy objects, environment monitoring, and surveillance. In this dissertation, we investigate the use of the method in some of these applications.

## 1.2 Problem Overview

Swarm robotics is a field in which we want to coordinate multi-robot systems composed of large numbers of mostly simple physical robots (Sahin and Winfield, 2008). In contrast with traditional multi-robot systems, which generally use centralized or hierarchical control and communication systems to coordinate robots' behaviors, swarm robotics adopts a decentralized approach in which the desired collective behaviors emerge from the local interactions between robots and their environment. Figure 1.1 presents a taxonomy illustrating the relationship between these research fields.

Swarm-based approaches offer several practical advantages over traditional robotics

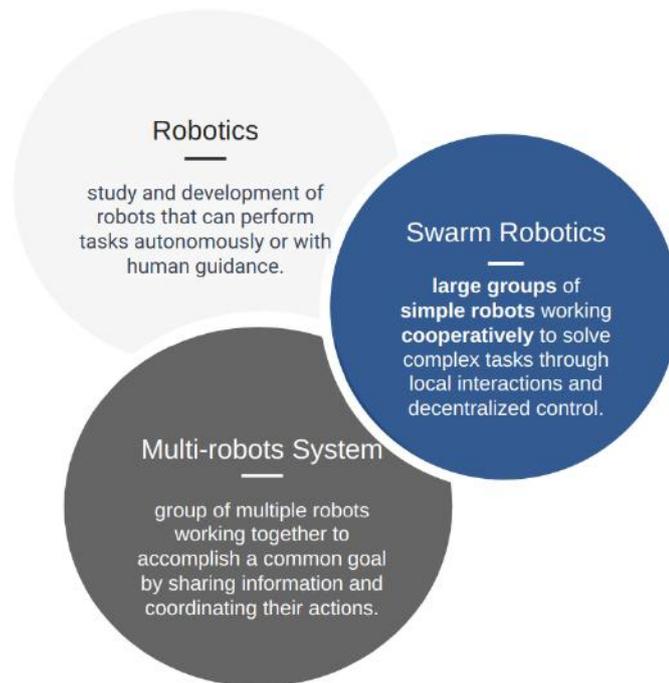


Figure 1.1: Taxonomy of swarm robotics: swarm robotics is categorized within the broader field of multi-robot systems, which in turn falls under the domain of robotics research.

systems, including enhanced robustness, flexibility, and scalability, as defined below.

**Definition 1.1** (Robustness). *Robustness is the capacity of the system to operate in the presence of partial failures or other abnormal conditions. A swarm is robust if they manage to complete the task even after losing several members or with considerable changes on the environment parameters.*

**Definition 1.2** (Flexibility). *Flexibility is the capability of the system to adapt to new, different, or changing requirements of the environment. Flexibility and robustness have partially conflicting definitions. The difference between the two occurs at the problem level. When the problem changes, the system has to be flexible (not robust) enough to switch to an appropriate behavior to solve the new problem. Biological systems have this flexibility and can easily switch their behaviors when problems change. For instance, ants are so flexible that they can solve foraging, prey retrieval, and chain formation problems with the same base self-organized mechanism (Tan and Zheng, 2013).*

**Definition 1.3** (Scalability). *Scalability is the ability of the system to expand a self-organized mechanism to support larger or smaller numbers of individuals without impact-*

*ing performance considerably. Although there is a range in which the swarm performs at acceptable performance levels, this range is preferred to be as large as possible.*

Although challenging, designing swarm robotics systems may provide powerful tools for a wide variety of applications in robotics. The problem tackled in this dissertation is to synthesize an effective solution to challenging swarm robotics problems with a minimalist approach. More specifically, we intend to create a novel methodology suitable for swarm robotics that allows synthesizing control strategies to produce different collective behaviors. The swarm robots we consider in this dissertation utilize reactive and decentralized control architectures relying exclusively on local sensing. In addition, we assume they have limited computational power, lack the capability for direct communication or networking, and are equipped with basic sensor units. More formally, the robots have the following constraints:

**Constraint 1.1** (Heterogeneity). *Robots may be homogeneous or heterogeneous.*

**Constraint 1.2** (Controllability). *Robots are reactive (act based on their current state).*

**Constraint 1.3** (Memorylessness). *Robots are memoryless (do not remember past states).*

**Constraint 1.4** (Leaderless). *No robot can be a leader or seed.*

**Constraint 1.5** (Communication). *Robots cannot directly communicate with each other.*

**Constraint 1.6** (Locality). *Robots only have access to their local state.*

**Constraint 1.7** (Environment). *Robots exist in a bounded space with obstacles.*

In order to design a minimalist solution, we consider the minimum amount of information required for robots to operate under the constraints above. Naturally, one may relax these strict conditions by adding other forms of sensing, increasing computation power and network capabilities to improve performance, allowing one to tackle more complex problems. For a minimalist approach, the following assumptions are made in this dissertation:

**Assumption 1.1** (Motion). *Robots are driven by a kinematic model with a known motion model. This is a reasonable assumption made in several swarm designs. In real robots, we assume this model may be imprecise and susceptible to noise, but it is still sufficient to describe most of the robot’s motion.*

**Assumption 1.2** (Sensing). *Robots can only sense obstacles and their neighbors’ relative positions and velocities up to a short range. We assume they can distinguish between obstacles and other robots as well as their types (heterogeneity). There exist several technologies that may be used to achieve such information. For instance, one may modulate infrared (IR) signals with specific codes to detect an obstacle or a robot, and if it is a neighbor robot, it can demodulate the signal to retrieve its types. Distance may be computed from the signal intensities and relative position given the sensor position regarding the robot frame. Relative velocity may be estimated by deriving the relative position. Other technologies may include audio (Basiri et al., 2014) and onboard vision-based methods (Faigl et al., 2013; Roelofsen et al., 2015).*

## 1.3 Objectives

Given the aforementioned problem overview, the objectives of this dissertation are twofold. First, inspired by concepts extensively used in statistical mechanics and quantum mechanics to model particle interactions, we aim to advance the state-of-the-art in swarm robotics by creating a novel methodology that allows synthesizing control strategies that produce different collective behavior using minimal sensing information. More specifically, we intend to extend the concepts of Gibbs Random Fields (GRFs) to the context of swarm robotics and use them to design decentralized control mechanisms that exploit local information to coordinate the behavior of a group of robots. Secondly, we investigate the flexibility of this methodology to address significant challenges in swarm robotics literature, as depicted in Figure 1.2. This entails designing methods that implicitly tackle flocking and segregation, cooperative object transportation, and pattern formation.

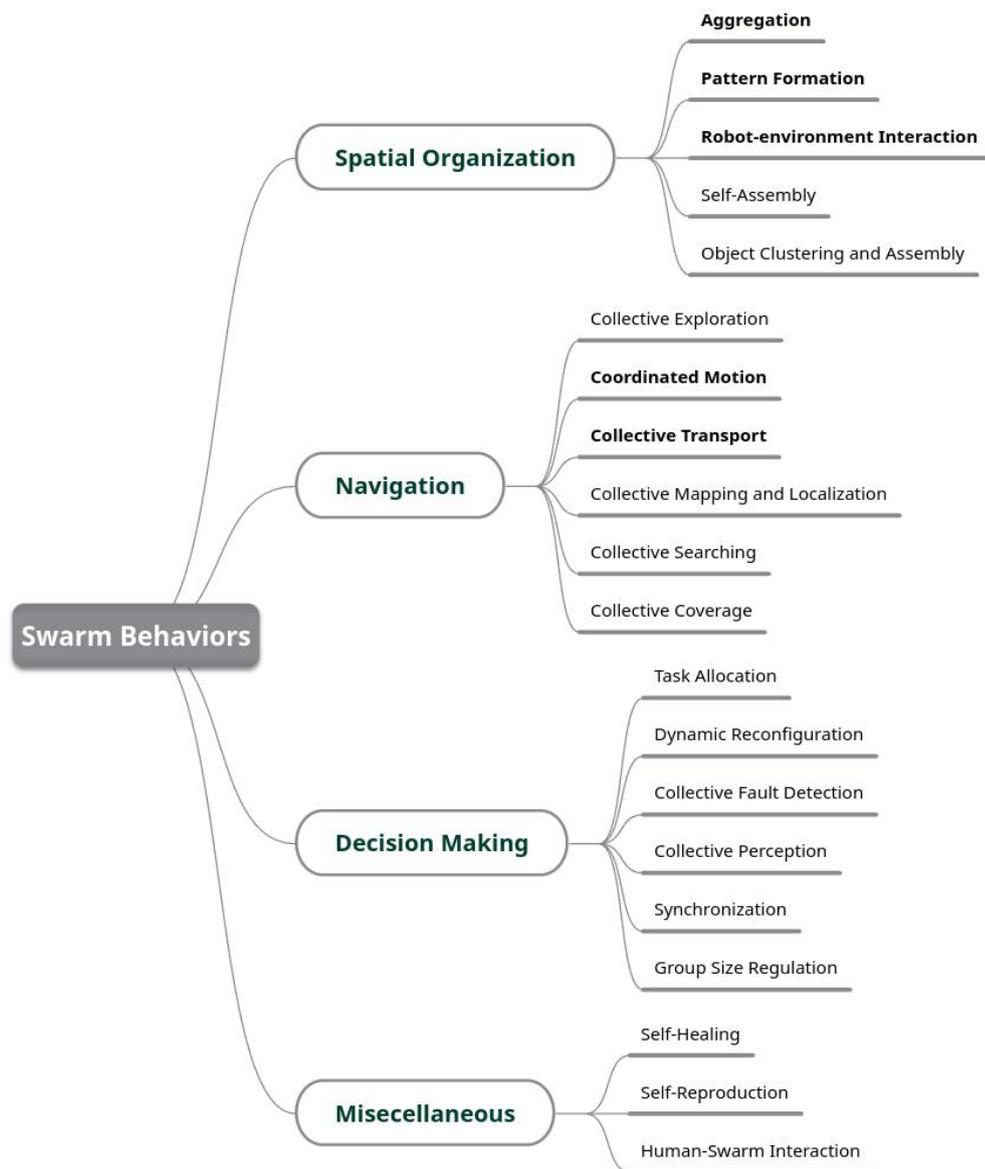


Figure 1.2: Swarm robotics behaviors adapted from [Schranz et al. \(2020\)](#). The highlighted behaviors are tackled in this dissertation.

Each of these problems has pertinent importance and potential applications in the real world. We first investigate the problem of integrating flocking with segregation, as they represent two fundamental capabilities that a robotic swarm must possess: form cohesive groups and navigate collectively ([Brambilla et al., 2013](#)). In this context, segregation is a particular type of aggregation behavior in which robots with common characteristics are placed together and set apart from other groups ([Santos et al., 2014a](#)). On the other hand, flocking is a coordinated motion behavior where the swarm moves in an arbitrary, organized, yet cohesive manner. In addition, we also incorporate robot-environment in-

teraction behavior, especially for obstacle avoidance, enhancing the swarm's adaptability and ability to navigate through intricate terrains or dynamic environments. Several applications can benefit from using a method that simultaneously performs these behaviors, such as area coverage, surveillance and reconnaissance, and foraging.

Robotic swarms capable of collectively transporting objects may also be suitable for many applications with high societal and economic impact potential. For instance, one may use robotic swarms for operations where the use of sophisticated robots is impossible or impractical, such as warehouse automation, waste disposal, and demining. Some of the main challenges consist of aligning and synchronizing the forces applied by the robots to sustain the transport of objects. Moreover, the method must be robust to objects of different shapes and resilient to changes in the environment, such as surfaces with different coefficients of friction and robot failures.

The pattern formation problem may be defined as the coordination of a group of robots to get into and maintain a formation with a certain shape (Bahceci et al., 2003). A key aspect for the applicability of these models in swarm robotics is the requirement for distributed and decentralized processing relying only on local information. Models with these characteristics bring several practical advantages allowing scalability, resiliency, and adaptability. Examples of potential applications would be oil spill containment or cleaning in oil plants (Kim et al., 2012; Shah et al., 2018) and constructing structures such as a temporary bridge that could dynamically adjust its size and shape to fit different environmental conditions (Rong et al., 2020).

Furthermore, we conduct a current state-of-the-art literature review of methods that tackle these swarm problems and computationally analyze the performance and capabilities of the control strategies through multiple numerical and physics simulations. Additionally, we evaluate the feasibility of the obtained control strategies in practice by demonstrating them in a physical autonomous differential-wheeled robotic swarm.

## 1.4 Contributions

This dissertation represents the author's work and includes a number of original contributions to scientific knowledge. The work presented herein has led so far to four publications in high-impact academic journals and conferences. The following states the contributions of this dissertation:

- A novel stochastic and decentralized approach that allows a swarm of heterogeneous robots to emerge with different behaviors relying entirely on local interactions with neighbors. The approach consists of modeling the swarm configuration as a dynamic Gibbs Random Field (GRF) and setting the GRF's potential energy as a combination of Coulomb-Buckingham potential and kinetic energy. We further formulate a probability distribution used by each robot to sample the most likely velocity commands. As a result, the robots can safely navigate through a bounded environment and interact with other robots emerging different swarm behaviors depending on the definition of an appropriate potential function. This approach is presented in Chapter 4 and is the basis of the three following contributions.
- The application of the methodology allows the design of a method that achieves simultaneously segregation and flocking behaviors using only local sensing. The method combines the Coulomb-Buckingham potential enabling the robots to aggregate and avoid obstacles in the environment, while using kinetic energy to allow the robots to reach a consensus on their relative velocities concerning their neighborhood. To the best of our knowledge, this is the first work to tackle these behaviors together, starting from a random initial state and using only local information. This method is elaborated upon in Chapter 5.

*Paulo Rezeck, Renato M. Assunção and Luiz Chaimowicz* **Flocking-Segregative Swarming Behaviors using Gibbs Random Fields**. 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 2021, pp. 8757-8763, [doi:10.1109/ICRA48506.2021.9561412](https://doi.org/10.1109/ICRA48506.2021.9561412). [Qualis-CC A1]

- Another example of application produces a method that allows a swarm of robots to navigate autonomously through the environment looking for the object to be transported, form groups, and push the object toward a goal location. These behaviors emerge from the local interactions without the need for explicit communication or coordination. The robots only need to be able to estimate their neighbors' relative position and velocity and distinguish between obstacles and objects detected within their sensing range. Moreover, the robots do not need any information about the object (i.e., location, size, mass, and shape), except for its relative goal location. The Coulomb-Buckingham potential enables the robots to aggregate, interact with the object by pushing it, and avoid obstacles in the environment. The kinetic energy allows the robots to reach a consensus on their relative velocities concerning their neighborhood and circulate the object looking for adequate pushing positions. This approach is presented in Chapter 6.

*Paulo Rezeck, Renato M. Assunção, Luiz Chaimowicz* **Cooperative Object Transportation using Gibbs Random Fields**. 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 2021, pp. 9131-9138, [doi:10.1109/IROS51168.2021.9635928](https://doi.org/10.1109/IROS51168.2021.9635928). [Qualis-CC A1]

- Another example is a method that allows a swarm of heterogeneous robots to emerge with interesting patterns relying entirely on local interactions with neighbors. The method extends the previous one by defining the neighborhood system inspired by the Octet rule used in chemistry. This approach has potential use in various scenarios, especially those where one may want to build more complex structures from simple ones. A possible application is modular robotics, in which complex robots

are built from simpler modules and can dynamically change their shape/structure. The bonding behaviors would guide the connection of the modules in a simple and dynamic fashion. Another application is the construction of temporary structures, such as bridges and platforms, with different industry and military uses. This approach is presented in Chapter 7.

*Paulo Rezeck and Luiz Chaimowicz* **Chemistry-Inspired Pattern Formation With Robotic Swarms**. 2022 IEEE Robotics and Automation Letters, vol. 7, no. 4, pp. 9137-9144, 2022, [doi:10.1109/LRA.2022.3190638](https://doi.org/10.1109/LRA.2022.3190638). [Qualis-CC A1]

- In addition to a novel methodology for generating different types of swarm behavior in a decentralized way using only local sensing, this dissertation also contributes to developing and improving the design and control of a swarm robotic platform called HeRo. The objective is to use the HeRo platform in real experiments to demonstrate the methodology's effectiveness. The novel design and feature improvement is presented in Appendix A.

*Paulo Rezeck, Héctor Azpúrua, Maurício F. S. Corrêa and Luiz Chaimowicz* **Hero 2.0: A Low-cost Robot for Swarm Robotics Research**. Autonomous Robots, p. 1-25, 2023, [doi:10.1007/s10514-023-10100-0](https://doi.org/10.1007/s10514-023-10100-0). [Qualis-CC A2]

## 1.5 Dissertation Outline

This dissertation is structured as follows:

- Chapter 2 gives a background on probabilistic graphical models, particularly for *Gibbs Random Fields* (GRFs). These include background concepts, definitions, and explanations of its properties and why they make sense in a swarm robotics context.
- Chapter 3 provides a review of the state-of-the-art approaches and techniques that

have been employed in the field of swarm robotics using Markov and Gibbs Random Fields, discussing how these probabilistic graphical models have been used to represent the interactions between robots and the environment. This chapter also highlights the contributions and novelty of our methodology.

- Chapter 4 presents a novel methodology for synthesizing minimalist models suitable for robot swarm control. We explain how we extend the concepts of GRFs to formulate a probability distribution used by each robot in a decentralized fashion to sample most likely velocity commands. By setting appropriate potential functions, the swarm produces different collective behaviors using minimal sensing information.
- Chapter 5 describes a complete study of a method for flocking and segregation behaviors. This includes reviews of the literature, details on the design of the method, numeric simulations demonstrating its capability, comparison with state-of-art approaches, and demonstration using physical differential-wheeled robots.
- Chapter 6 describes a complete study of a method for cooperative object transportation using the pushing strategy. We review the literature, detail the method, demonstrate its scalability, adaptability, and robustness using physical simulations, and show proof-of-concept experiments using physical robots.
- Chapter 7 describes a complete study of a method for pattern and shape formation behavior. We review the most relevant works, detail the method, demonstrate its versatility in producing different patterns, and show physical simulations and real-robot experiments in which a swarm was capable of building a chain-like structure, which is potentially attractive for more tangible applications.
- Chapter 8 outlines the key conclusions drawn from this dissertation and presents insights into potential future research directions.

# Chapter 2

## Background

This chapter presents an overview of some concepts about *Gibbs Random Fields* (GRFs), explaining their properties and why they make sense in a swarm robotics context. Originally, they have been broadly used in solving various problems in statistical mechanics but have acquired a strongly interdisciplinary character, with applications to neural network theory, computer science, theoretical biology, and economy. In this dissertation, we investigate its use as a swarm control mechanism.

### 2.1 Random Fields and Markov Properties

A Random Field (RF) is a generalization of a stochastic process where the underlying parameter takes values from a multidimensional space instead of being one-dimensional, typically temporal (Vanmarcke, 2010).

**Definition 2.1** (Random field). *Formally, given a probability space  $(\Omega, \mathcal{X}, P)$ , a random field  $\mathbf{X}$  is a collection of random variables indexed by elements in a topological space  $T \subset \mathbb{R}^n$ . That is,*

$$\mathbf{X} = \{X_u\}_{u \in T}.$$

Some examples of random fields are the Markov Random Field (MRF), Gibbs Random Field (GRF), Conditional Random Field (CRF), and Gaussian Markov Random Field (GMRF), and in this dissertation, we will concentrate on the first two.

A random field is said to be a MRF if it satisfies Markov Properties. The term Markov property refers to the memoryless property of a stochastic process and allows one to ignore more distant information as soon as local information is provided. More specifically, an MRF satisfies the following Markov properties (Koller and Friedman, 2009):

**Definition 2.2** (Pairwise Markov property). *Any two non-neighbor variables  $\{u, v\}$  are conditionally independent given all other variables  $T \setminus \{u, v\}$ . That is,*

$$X_u \perp\!\!\!\perp X_v \mid X_{T \setminus \{u, v\}},$$

*and implies that the probability of the hypothesis given the uninformative observation  $X_{T \setminus \{u, v\}}$  is equal to the probability without it, which means that there is no information gain when observing the entire state of the process.*

**Definition 2.3** (Local Markov property). *A variable  $v$  is conditionally independent of all other variables given its neighbors. That is,*

$$X_v \perp\!\!\!\perp X_{T \setminus \mathcal{N}_{v+}} \mid X_{\mathcal{N}_v},$$

*where  $\mathcal{N}_v$  is the set of neighbors of  $v$ , and  $\mathcal{N}_{v+} = v \cup \mathcal{N}_v$  is the closed neighborhood of  $v$ . It implies that only the observation of the neighbors is relevant, and it is unnecessary to observe the whole process.*

**Definition 2.4** (Global Markov property). *Any two subsets of variables  $\{A, B\}$  are conditionally independent given a separating subset  $S$ . That is,*

$$X_A \perp\!\!\!\perp X_B \mid X_S,$$

*where a separating subset  $S$  is one in which every path from a subset  $A$  to a subset  $B$  passes through it. It implies that observing variables on a boundary (not necessarily neighbors) is redundant to information coming from variables beyond that boundary.*

**Remark.** *Note that such properties may be convenient to model swarm robots since they imply the independence of information coming from outside a neighborhood system, which supports the requirement of local interaction.*

## 2.2 Markov and Gibbs Random Fields

A MRF is a mathematical model that represents the joint distribution of a set of random variables that are defined on a graph. The variables are assumed to have the Markov Properties described in the previous section, meaning that the distribution of any variable is conditionally independent of all other variables given its neighbors in the graph. This allows the joint distribution to be written as a product of local distributions, making it easier to study the system.

The Hammersley-Clifford theorem (Kindermann, 1980) is a fundamental result in the study of Markov Random Fields (MRFs) and Gibbs Random Fields (GRFs). This theorem establishes the equivalence between these two models, commonly used in statistical mechanics and machine learning to describe the distribution of system configurations.

A GRF is a special case of a MRF that satisfies certain additional conditions. In particular, the joint probability density of the random variables must be strictly positive and also satisfy the Markov Properties of a MRF. The Hammersley-Clifford theorem states that any GRF can be represented as a MRF, and vice versa. This means that researchers can use either model to study a given system, depending on the specific assumptions and computational techniques that are most appropriate for the problem at hand.

To formally describe these models, let us assume the topological space as an undirected graph  $\mathbf{G} = (\mathbf{V}, \mathbf{E})$  with vertices as spatial sites and indexed by  $v = 1, 2, \dots, \eta$ . A random field on  $\mathbf{G}$  is a collection of random variables

$$\mathbf{X} = \{X_v\}_{v \in \mathbf{V}}$$

and, for each  $v \in \mathbf{V}$ , let  $\Lambda_v$  be finite set called the phase space for site  $v$  that represents where the random variable  $X_v$  takes its values.

An instance of  $\mathbf{X}$  establishes a state of the random field

$$x = \{(x_1, \dots, x_\eta) : x_v \in \Lambda_v, v \in \mathbf{V}\}$$

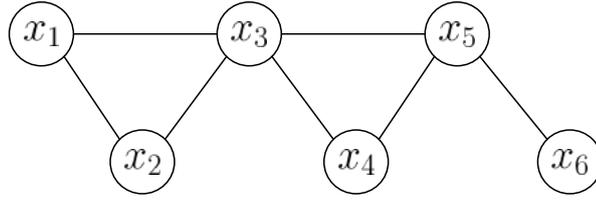


Figure 2.1: An instance of a random field with  $\mathbf{X} = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ .

and the product space

$$\Lambda \triangleq \Lambda_1 \times \dots \times \Lambda_\eta$$

forms the configuration space.

Figure 2.1 gives an example where edges connote some correlation between the connected vertices. That is, the contribution of a single site to the whole is made through immediate interactions with its neighboring sites.

**Definition 2.5** (Neighborhood System). *A neighborhood system on  $\mathbf{V}$  is a family*

$$\mathcal{N} = \{\mathcal{N}_v\}_{v \in \mathbf{V}},$$

where  $\mathcal{N}_v \subset \mathbf{V}$  is the set of neighbors for site  $v$  satisfying

$$v \notin \mathcal{N}_v$$

and the symmetry

$$r \in \mathcal{N}_v \Leftrightarrow v \in \mathcal{N}_r.$$

The neighborhood system induces the configuration of the undirected graph  $\mathbf{G}$  by setting an edge  $\{v, r\} \in \mathbf{E}$  between  $v$  and  $r$  if and only if  $r \in \mathcal{N}_v$  (e.g.  $\mathcal{N}_3 = \{x_1, x_2, x_4, x_5\}$  in Figure 2.1).

By assuming the local Markov Property (2.3), given its neighbors  $\mathcal{N}_v$ , a site  $v$  is independent of all other sites in the graph. Therefore, we may write the following conditional probability,  $\forall v \in \mathbf{V}$

$$P(\mathbf{X}_v = x_v \mid (\mathbf{X}_r = x_r)_{r \neq v}) = P(\mathbf{X}_v = x_v \mid (\mathbf{X}_r = x_r)_{r \in \mathcal{N}_v}), \quad (2.1)$$

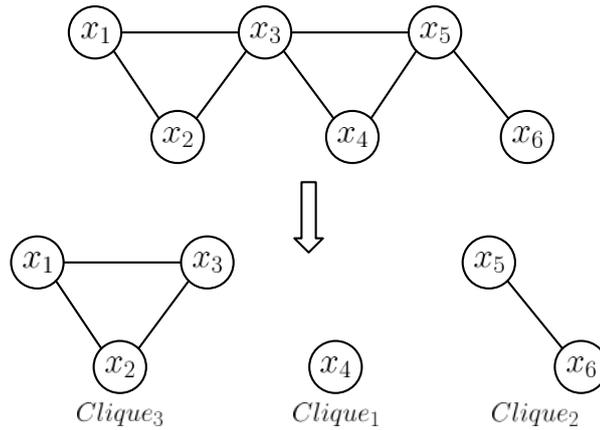


Figure 2.2: An instance of clique problem for a graph with a maximum clique size of 3.

which indicates that the conditional probability of the site  $v$  assuming the state  $x_v$  given the state of all other sites is equal to the probability of  $v$  assuming the same state  $x_v$  given only the states of neighboring sites.

Moreover, the global Markov property (2.4) tells us that the joint distribution of  $\mathbf{X}$  is determined entirely by the local conditional distributions. But it is unclear how to actually construct the global joint distribution from these local functions. In order to do this, we need to factorize according to the *cliques* of the graph.

**Definition 2.6** (Clique). *Cliques are associated with neighborhood systems such that a subset  $C \subseteq \mathbf{V}$  is called a clique if all elements of  $C$  are neighbors of each other. That is, a clique is any fully connected subset of the graph (e.g.,  $\{x_4\}$ ,  $\{x_5, x_6\}$  or  $\{x_1, x_2, x_3\}$  in Figure 2.2). A maximal clique is a clique that cannot be extended by the addition of other elements. We denote the set of all maximal cliques in the graph as a family  $\mathcal{C}$ , with a clique  $C \in \mathcal{C}$  comprising the maximum number of vertices.*

**Definition 2.7** (Markov Random Field). *Given a set of random variables  $\mathbf{X} = \{X_v\}_{v \in \mathbf{V}}$ , let  $P(\mathbf{X} = x)$  be the probability of a particular field configuration  $x$  in  $\mathbf{X}$ . That is,  $P(\mathbf{X} = x)$  is the probability of finding that the random variables  $\mathbf{X}$  take on the particular value  $x$ . If this joint density can be factorized over the cliques of the graph  $\mathbf{G}$ , then we may rewrite the conditional probability (2.1) as a joint density,*

$$P(\mathbf{X} = x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} U_C(x_C) \quad (2.2)$$

and then  $\mathbf{X}$  forms a Markov random field with respect to  $\mathbf{G}$ . The functions  $U_C(\cdot)$  are referred to as factor potentials or clique potentials of a configuration  $x_C$ , and  $Z$  is called the partition function, and takes the form,

$$Z = \sum_{x \in \Lambda} \prod_{C \in \mathcal{C}} U_C(x_C).$$

**Remark.** Note that the conditional probabilities (2.1) are subject to non-obvious and highly restrictive consistency conditions, such that one may factorize them in terms of the joint probability (2.2). In the context of swarm robotics, this property shows that the global state of the swarm may be explained only using local observations.

As we mentioned, a GRF is a particular case of the MRF. More specifically, it is a particular application for the joint probability density (2.2), which uses the Gibbs measure. A Gibbs measure is a generalization of the canonical ensemble to infinite systems, which gives the probability of the system  $\mathbf{X}$  being in the state  $x$  by measuring the system's energy. To properly define the GRF, we must formally define the concept of potentials function presented in the definition (2.7).

**Definition 2.8** (Potential Energy). Let us denote a potential  $\mathbf{U}$  as a family  $\{U_A : A \subset \mathbf{V}\}$  of functions on the configuration space  $\Lambda$ , where  $U_A : \Lambda \rightarrow \mathbb{R}$ , and  $U_A(x)$  depends only on  $x_A \triangleq \{x_v : v \in A\}$ . That is,  $U_A$  is only a function of the values at the sites contained in the set  $A$ , that is  $U_A(x) \equiv U_A(x_A)$ . In this way, given a potential  $\mathbf{U}$ , the potential energy  $H(x)$  for configuration  $x$  is defined as

$$H(x) = \sum_{A \subset \mathbf{V}} U_A(x_A). \quad (2.3)$$

**Definition 2.9** (Gibbs Random Field). Finally, a random field  $\mathbf{X}$  is called a GRF if,

$$P(\mathbf{X} = x) = \frac{1}{Z} e^{-\frac{H(x)}{T}}, \quad \text{with } Z = \sum_z e^{-\frac{H(z)}{T}}, \quad (2.4)$$

where  $Z$  is the partition function (normalizing constant);  $T$  is interpreted as temperature in the context of statistical physics; and  $\frac{1}{Z} e^{-\frac{H(x)}{T}}$  is called Gibbs distribution or Boltzmann distribution.

---

The Gibbs Random Field (GRF) is a mathematical model that is commonly used in statistical mechanics and mathematics to describe the distribution of system configurations at thermodynamic equilibrium. This model is often used to measure the probability of a system yielding a desired state, and it is typically represented by the Gibbs distribution in (2.4). One of the challenges of directly evaluating this equation is the high cardinality of the configuration space, which makes the computation of the normalizing constant  $Z$  intractable in practice. This can be a major hurdle for researchers who want to use the GRF to study complex systems, such as large molecular systems or networks of interacting agents.

One way to overcome this challenge is to use approximate inference methods, such as Markov Chain Monte Carlo (MCMC) methods, to sample from the GRF. These methods allow researchers to generate samples from the distribution without explicitly computing the normalizing constant. This can be a more efficient and scalable approach for studying systems with a large configuration space, and it has been widely used in a variety of scientific and engineering applications. A process to parallelly sample over (2.4) is described as the methodology for this dissertation.

# Chapter 3

## Related Work

This chapter presents a review of the literature on the use of Markov and Gibbs Random Fields in the context of swarm robotics. Works related to the three applications of our methodology are presented later in the dissertation. Chapter 5 presents works related to segregation and flocking, while Chapter 6 focuses on cooperative transportation, and Chapter 7 covers pattern formation behavior.

Markov Random Fields (MRFs) and Gibbs Random Fields (GRFs) are mathematical models that can be used to describe the behavior of systems that consist of many individual components that interact with each other. These models are often used in statistical mechanics to study the behavior of large collections of particles, such as gases or liquids (Kindermann, 1980). In image processing and computer vision, they are used to model the interactions between the pixels in an image, allowing algorithms to make more accurate predictions about the content of an image (Blake et al., 2011).

In the context of swarm robotics, Markov and Gibbs random fields could potentially be used to model the interactions between individual robots in a swarm. By representing each robot as a node in a MRF or GRF, it would be possible to use these models to predict the collective behavior of the swarm based on the interactions between the individual robots. This could be useful for tasks such as swarm formation, collective decision-making, and coordination of multi-robot systems. Despite the potential of these models for swarm robotics, there are relatively few works in the literature that describe their applications in this context, which highlights the need for more research in this area.

Baras and Tan (2004) provide a valuable contribution to the field of swarm robotics and demonstrate the potential of using MRFs to model and control the behavior of swarms.

---

The authors model a swarm of homogeneous robots as a MRF on a graph, where the robots and their sensing links constitute the vertices and the edges of the graph, respectively. Global objectives, such as gathering at one place or dispersion for maximum area coverage, are reflected through the design of Gibbs potentials, which define the interactions between the robots in the MRF. To control the movement of the robots, the authors used simulated annealing based on the Gibbs sampler. This allowed the robots to move in a way that was consistent with the Gibbs potentials and the global objectives of the swarm. Because the Gibbs potentials consist of locally coupling terms, the computation of each robot's next move only requires information about its neighbors on the graph. This makes the approach scalable and efficient, even for large swarms of robots. The authors first described a sequential sampling approach, where the robots take turns updating their locations based on the Gibbs sampler. They also discussed an extension to parallel sampling, where each robot computes its next move simultaneously, and evaluated the performance of this approach in terms of the speed and accuracy of the swarm's movement.

[Xi et al. \(2006\)](#) propose a hybrid algorithm that combines elements of gradient descent and Gibbs sampling to achieve efficient optimization in multi-robot systems. The algorithm uses a two-step approach, in which the robots first move based on a proposal distribution that is designed to take into account the characteristics of the environment. In the second step, the robots make a move based on the local characteristics of the Gibbs distribution. This allows the algorithm to take advantage of the global objectives of the system while avoiding getting stuck at local minima. The authors show that this two-step approach can lead to convergence to minimal potential configurations, and they also analyze the impact of different switching parameters and the role of robots' memory on performance enhancement. Further, the authors also investigate the robustness of their algorithm in the presence of uncertainties in sensing ([Xi and Baras, 2006](#)). They study two types of potential errors: range errors and random errors. They show that if the bound of the range errors are less than half of the step size of the algorithm, the annealing algorithm will still converge to the same configuration as in the case where there are no errors. This suggests that the algorithm is relatively robust to errors in sensing, as long

---

as the errors are not too large.

Previous works have demonstrated the potential of using Markov and Gibbs random fields to optimize multi-robot systems' behavior. However, these works have not formally explained why the algorithms work and under what conditions they are likely to be effective. This is a challenging problem because the neighborhood system, or the information graph, in a swarm of robots can vary with the configuration of the swarm. This differs from classical Markov random fields, where the neighborhood system is assumed to be fixed (Kindermann, 1980; Winkler, 2003). Additionally, the parallel sampling approach used in the context of swarm robotics poses additional challenges, as it involves all robots updating their locations simultaneously rather than sequentially. Tan et al. (2010) address these challenges by presenting a rigorous analysis of a parallel Gibbs sampling-based algorithm for coordinating the behavior of multi-robot systems. The proof is based on pairwise potentials, which consist of contributions from pairs of nodes in the system. The authors establish that under a constant temperature for the Gibbs distribution, the parallel sampling algorithm results in a unique stationary distribution for the swarm configuration. They also show that if the temperature follows an appropriate annealing schedule, the configuration will converge to the global minimizers of a modified potential energy function. This work provides insight into why these algorithms work and under what conditions they are likely to be effective. Moreover, the authors note that the algorithm and the proof do not explicitly require the connectivity of the graph.

Fernando (2021) presents a recent flocking algorithm that combines differentially flat dynamics with Markov Random Fields (MRFs) to model the interactions among robots in a swarm. The author uses a Self-Propelled Particle (SPP)-based energy function to define a domain for the random variables induced by the robots' control actions, and they propose using variational inference based on the Mean-Field Approximation (MFA) to approximate the best control input that minimizes the energy in the neighborhood of each robot. The use of MFA allows for fast, computationally tractable inference, which is crucial for the online coordination and collision avoidance that is necessary for the flocking algorithm to work effectively.

---

The work presented in this dissertation has been inspired by [Baras and Tan \(2004\)](#) that used MRFs to self-organize a swarm of homogeneous robots. Besides modeling the swarm heterogeneity, there are other crucial differences between this work and theirs. First, our approach allows for the incorporation of robot kinematics, allowing us to model the continuous movement of the robots in a bounded environment, whereas they consider a discrete and a bounded environment and assume lattices as their environment. Second, we introduce a different potential function that enables us to encode low-level swarm behaviors and this makes all the difference in our approach. Specifically, we adopt the Coulomb-Buckingham Potential ([Buckingham, 1938](#)) coupled with a Kinetic Energy term to model the robots' interactions. Finally, we also demonstrate the versatility of our methodology by applying it to different swarm tasks, including proof-of-concept experiments using real robots.

Especially, we present a method that achieves both segregation and flocking behaviors simultaneously, using the Coulomb-Buckingham potential to enable robots to aggregate and avoid obstacles, while using kinetic energy to allow robots to reach a consensus on their relative velocities. Another example is a method that enables a swarm of robots to navigate autonomously, form groups, and push an object toward a goal location, without the need for explicit communication or coordination. The method utilizes the Coulomb-Buckingham potential to enable robots to aggregate, interact with the object by pushing it, and avoid obstacles, while using kinetic energy to allow robots to reach a consensus on their relative velocities. Another example is a method that allows a swarm of heterogeneous robots to emerge with interesting patterns relying entirely on local interactions with neighbors. This approach has potential use in various scenarios, such as modular robotics, where complex robots are built from simpler modules that dynamically change their shape/structure. The successful application of our methodology to these distinct tasks underscores its flexibility and potential as a general-purpose approach for swarm control and coordination.

# Chapter 4

## Methodology

In this chapter, we extend the GRFs concepts to the context of swarm robotics and present our modeling and control strategy. The use of GRFs in swarm robotics is a relatively novel approach to modeling and controlling the behavior of a swarm of robots. In a nutshell, a GRFs is a mathematical model that can describe the spatial variation of a physical quantity over a continuous region. In the context of swarm robotics, this physical quantity might be the configuration of the robots in the swarm, with the spatial variation representing the positions of the robots over time.

The idea behind using GRFs in swarm robotics is to model the swarm's behavior as a random field, where the global minimum of the potential represents the desired behavior or configuration for the swarm. By sampling velocities for each robot in a decentralized way, the robots can move toward the global minimum of the potential, leading the entire swarm to converge toward the desired behavior. This approach has several advantages over more traditional methods for modeling and controlling the behavior of a swarm. Firstly, it allows for decentralized control, where each robot makes decisions based on local information rather than relying on a central controller. This makes the system more robust and scalable, as there is no single point of failure, and the swarm can continue to operate even if individual robots fail. Additionally, using GRFs allows for a more flexible and adaptable approach to swarm behavior, as the potential field can be easily modified to account for changing environmental conditions or new goals for the swarm.

## 4.1 Formalization

To begin, let us assume a set  $\mathcal{R}$  of  $\eta$  heterogeneous robots, initially distributed in a random configuration with unknown topology and moving in a bounded region within the two-dimensional Euclidean space<sup>1</sup>. The objective is to find appropriate velocity commands for each robot in a decentralized way that leads the entire swarm to converge toward the global minimum of the potential. Prior to delving into the specifics of our methodology, let us establish key concepts and underlying assumptions.

**Definition 4.1** (Robot State). *The state of the  $i$ -th robot at time step  $t$  is represented exclusively by its instantaneous velocity vector  $\mathbf{v}_i^{(t)}$ , which is bounded by robot's maximum speed  $v_{max}$ , that is  $\|\mathbf{v}_i^{(t)}\| \leq v_{max}$ . The robot's instantaneous velocity  $\mathbf{v}_i^{(t)}$  refers to the current velocity of a robot at a specific point in time and concerning its body reference frame, as depicted in Figure 4.1. It describes both the magnitude and direction of the robot's body motion at that instant  $t$ .*

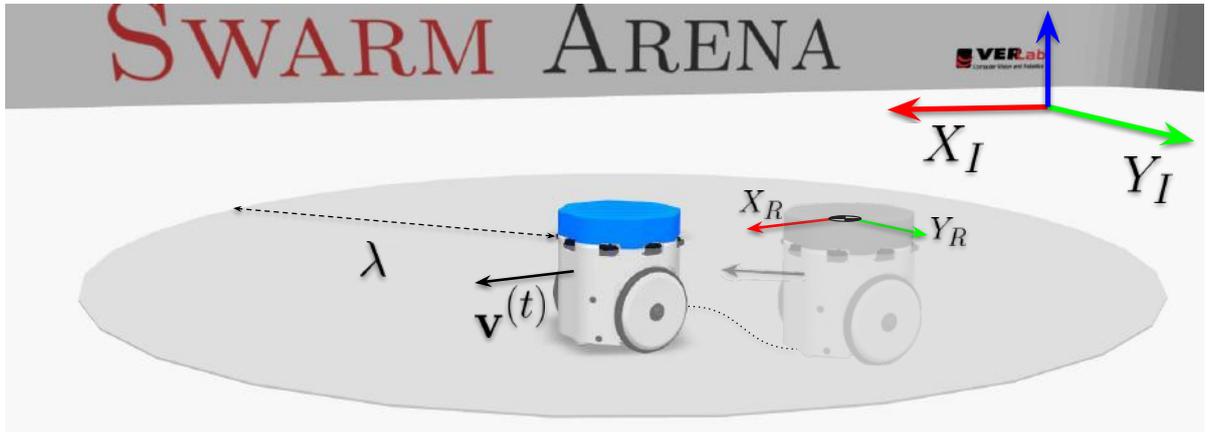


Figure 4.1: Robot state representation: the state of the robot is exclusively defined by its instantaneous velocity  $\mathbf{v}^{(t)}$  with respect to its previous body reference frame  $\{\mathbf{X}_R, \mathbf{Y}_R\}$ . This implies that the robot lacks knowledge of any reference to the inertial or world frame  $\{\mathbf{X}_I, \mathbf{Y}_I\}$ . The circular gray region illustrates the robot's sensing range, wherein it can measure distances from obstacles or other robots, as well as estimate their relative velocities. This sensing range is constrained by a maximum distance  $\lambda$ .

<sup>1</sup>This assumption can straightforwardly extend to a three-dimensional space.

**Definition 4.2** (Swarm Configuration). *The swarm configuration at time step  $t$  is defined as a set  $\mathbf{x}^{(t)}$ <sup>2</sup> comprising the instantaneous velocity vectors of all the robots, expressed as*

$$\mathbf{x}^{(t)} = \{\mathbf{v}_1^{(t)}, \mathbf{v}_2^{(t)}, \dots, \mathbf{v}_\eta^{(t)}\}.$$

**Definition 4.3** (Heterogeneity). *The swarm heterogeneity is modeled by a partition  $\tau = \{\tau_1, \dots, \tau_u\}$ ,  $u \leq \eta$ , with each  $\tau_k \subset \mathcal{R}$  containing exclusively all robots of type  $k$ , i.e.  $\forall(j, k) : j \neq k \rightarrow \tau_k \cap \tau_j = \emptyset$ . The heterogeneity is defined by the mass  $m$  and electrical charge<sup>3</sup>  $c$  of the robot so that,*

$$\forall(i, j) \in \tau_k : \tau_k \subset \tau \rightarrow m_i \triangleq m_j \text{ and } c_i \triangleq c_j.$$

*That is, robots of the same type share the same mass ( $m$ ) and electrical charge ( $c$ ) values. By manipulating these parameters, one may model how the robots interact with each other and design specific behaviors depending on their type.*

**Assumption 4.1** (Sensing). *The robots have a circular sensing range of radius  $\lambda$ , as depicted in Figure 4.1. Within this radius, the robot estimates its distance in line of sight from obstacles in the environment as well as the distance from other robots. Additionally, we assume the sensors can estimate the relative instantaneous velocities of other robots as well as their types. Formally, the  $i$ -th robot uses its sensors to estimate the relative position  $\mathbf{p}_{ij}^{(t)}$ , relative instantaneous velocity  $\mathbf{v}_{ij}^{(t)} \triangleq \mathbf{v}_j^{(t)} - \mathbf{v}_i^{(t)}$  and type,  $\forall j \in \mathcal{R}$  that is within the sensing range. The sensor is also capable of differentiating between robots and obstacles and can estimate the distance for a set of sampling points outlining the obstacle.*

**Assumption 4.2** (Motion Model). *We assume that the motion model of all robots is known. That is, there is a mathematical model that describes how each robot moves by taking its instantaneous velocity for a time interval. Formally, we manipulate typical representation for the motion model to predict how far robots move away from each other. More specifically, the  $i$ -th robot predicts how far a  $j$ -th robot moves away for a time interval using the following motion model:*

$$\mathcal{K} : (\mathbf{p}_{ij}^{(t)}, \mathbf{v}_{ij}^{(t)}) \rightarrow (\mathbf{p}_{ij}^{(t+1)}).$$

<sup>2</sup>From now on, we use the symbol  $\mathbf{x}$  to represent robot velocities.

<sup>3</sup>The electrical charge is a parameter with no tangible concept in this context of swarm robotics.

**Definition 4.4** (Obstacles). *The obstacles within the environment are represented as a finite set of points  $\mathcal{W} = \{\mathbf{w}_1, \dots, \mathbf{w}_n\}$  outlining its perimeter. An obstacle detected by the  $i$ -th robot at time step  $t$  consists of a subset of points  $\mathcal{W}_i^{(t)} \subset \mathcal{W}$  in the robot frame, where*

$$\mathbf{w}_j \in \mathcal{W}_i^{(t)} \rightarrow \|\mathbf{w}_j - \mathbf{p}_i\| \leq \lambda,$$

and  $\|\mathbf{w}_j - \mathbf{p}_i\|$  is the Euclidean distance between the  $i$ -th robot and  $j$ -th point.

**Definition 4.5** (Neighborhood System). *The neighborhood system (previously defined in 2.5) applied for the  $i$ -th robot at time step  $t$  and constrained by the sensing range  $\lambda$ , defines a set of neighboring robots, where*

$$\mathcal{N}_i^{(t)} \triangleq \{j \in \mathcal{R} : j \neq i, \|\mathbf{p}_{ij}^{(t)}\| \leq \lambda\},$$

where  $\mathbf{p}_{ij}^{(t)}$  is the relative position and  $\|\mathbf{p}_{ij}^{(t)}\|$  is the Euclidean distance between the  $i$ -th and  $j$ -th robots at time step  $t$ .

**Remark.** *Note that the neighborhood definition only requires the Euclidean distance between the two robots. That is, it requires only a sensor that estimates the relative position between the robots, and it is not necessary to maintain any form of localization or global information. Additionally, it is important to highlight that the neighborhood may dynamically change over time as the robots interact with one another. This implies that we do not assume any predefined topology for the neighborhood system.*

## 4.2 Modeling Robotic Swarms using GRFs

In the context of swarm robotics, GRFs can be used to model the local interactions between the robots in the swarm. Expanding upon the modeling introduced in Chapter 2, we define the swarm configuration at time step  $t$  as a graph denoted by  $\mathbf{G}^{(t)} = (\mathcal{R}, \mathbf{E}^{(t)})$  with fully dynamic connectivity that evolves over time<sup>4</sup>.

<sup>4</sup>This implies that the graph may be disconnected or connected depending on the robot's interactions.

A neighborhood system on  $\mathcal{R}$ , given the robots sensing information at time step  $t$ , is a family  $\mathcal{N}^{(t)} = \{\mathcal{N}_i^{(t)}\}_{i \in \mathcal{R}}$ , where  $\mathcal{N}_i^{(t)} \subset \mathcal{R}$  is the set of neighbors defined in (4.5) and satisfies  $i \notin \mathcal{N}_i^{(t)}$  and the symmetry  $j \in \mathcal{N}_i^{(t)} \Leftrightarrow i \in \mathcal{N}_j^{(t)}$ . The neighborhood system  $\mathcal{N}^{(t)}$  induces the configuration of the graph  $\mathbf{G}^{(t)}$  by establishing an edge  $(i, j) \in \mathbf{E}^{(t)}$  between the  $i$ -th and  $j$ -th robots if and only if  $j \in \mathcal{N}_i^{(t)}$ .

**Remark.** Note that the neighborhood system is not inherently static, but it can vary as the swarm adapts its configuration to exhibit a certain swarm behavior. This differs from classical Gibbs random fields, where the neighborhood system is assumed to be fixed. Therefore, we use the term **dynamic** Gibbs Random fields to imply that the neighborhood system is evolving over time, leading to dynamic connectivity within the graph at initial time steps, eventually converging to a stable topology, as depicted in Figure 4.2.

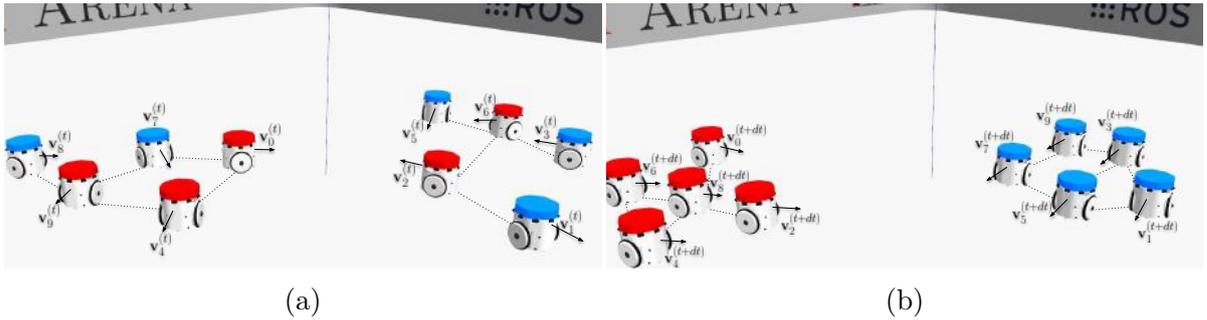


Figure 4.2: Neighborhood system evolving over time: (a) neighborhood system at time step  $t$ ,  $\mathcal{N}^{(t)}$ , induces the graph  $\mathbf{G}^{(t)}$ . After a sufficiently long time interval  $dt$ , the neighborhood system  $\mathcal{N}^{(t+dt)}$  converges to consistent topology inducing the graph  $\mathbf{G}^{(t+dt)}$ . Depending on the swarm behavior, the graph may converge to different topologies. For instance, in the aggregation behavior, where all robots must be as close as possible to each other, the graph should be connected. Conversely, for dispersion behavior, it is expected that the graph converges to a fully disconnected topology.

Additionally, let us define a random field in  $\mathbf{G}$  by introducing a set of random variables  $\mathbf{X} = \{X_i\}_{i \in \mathcal{R}}$ . A random variable  $X_i$  represents the instantaneous velocities of the  $i$ -th robot, and  $\Lambda_i$  is a finite set called the phase space that encompasses all possible velocity vectors that  $X_i$  may assume. Formally,  $\Lambda_i$  is a connected (continuous) sample space bounded by the robot's maximum velocity, that is,  $\forall \mathbf{v}_i \in \Lambda_i \rightarrow \|\mathbf{v}_i\| \leq v_{max}$ . An instance of  $\mathbf{X}$  establishes a state of the random field and represents a specific swarm configuration  $\mathbf{x} = \{\mathbf{v}_1, \dots, \mathbf{v}_\eta\}$ , essentially the velocities executed by each robot.

After establishing our approach to modeling the swarm as a random field, let us further expand upon the concepts of potential energy. Referring to the definition of potential energy (2.8),

$$H(x) = \sum_{A \subset \mathbf{V}} U_A(x_A),$$

one may note that *cliques* (2.6) represent the range of the interactions that contribute to the total energy of the whole system. In addition to the potential energy definition, two other definitions may allow one to decompose the potential energy (2.3) into two terms for a more tangible application for swarm robotics.

**Definition 4.6** (Pairwise, Nearest-Neighbor Potential). *if  $U_A \equiv 0$  whenever  $A$  is not a clique or a singleton,  $\mathbf{U}$  is called a nearest-neighbor potential. If  $U_A \equiv 0$  whenever  $A$  is not a pair or a singleton,  $\mathbf{U}$  is called a pairwise potential.  $\mathbf{U}$  is called a pairwise, nearest-neighbor potential if it is both a pairwise potential and a nearest-neighbor potential.*

In particular, for a pairwise, nearest-neighbor potential  $U$ , we can rewrite (2.3) as

$$H(x) = \sum_{i \in \mathcal{R}} U_{\{i\}}(x_i) + \sum_{(i,j) \in \mathcal{R} \times \mathcal{R}, j \in \mathcal{N}_i} U_{\{i,j\}}(x_i, x_j). \quad (4.1)$$

By incorporating these definitions into the GRF model, one may rewrite the joint distribution (2.4) as

$$P(\mathbf{X} = x) = \frac{1}{Z} e^{-\frac{\sum_{i \in \mathcal{R}} U_{\{i\}}(x_i) + \sum_{(i,j) \in \mathcal{R} \times \mathcal{R}, j \in \mathcal{N}_i} U_{\{i,j\}}(x_i, x_j)}{T}}, \quad (4.2)$$

with

$$Z = \sum_z e^{-\frac{\sum_{i \in \mathcal{R}} U_{\{i\}}(z_i) + \sum_{(i,j) \in \mathcal{R} \times \mathcal{R}, j \in \mathcal{N}_i} U_{\{i,j\}}(z_i, z_j)}{T}},$$

where  $Z$  is the partition function (normalizing constant) that accounts for the potential of all possible  $z$  configurations;  $T$  is interpreted as the temperature in the context of statistical physics and taken as equal to 1 from now on; and  $\frac{1}{Z} e^{-\frac{H(x)}{T}}$  is called Gibbs distribution or Boltzmann distribution.

Note that these definitions only allow us to calculate the probability of the entire swarm reaching a certain configuration  $x$ . But what we require is to sample velocities for each robot given the information about the robots in its neighborhood. Next, we explain how we perform such a procedure in a decentralized way using Gibbs sampling.

### 4.3 Parallel Gibbs Sampling

In parallel Gibbs sampling, each robot samples velocities simultaneously based on the configuration  $\mathbf{x}$  and the local Gibbs potential energy. This is possible because the Gibbs potential energy is a local quantity, meaning that it only depends on the neighborhood around the robot. This allows each robot to update its instantaneous velocity independently of the other robots, making parallel sampling possible.

Formally, let  $\mathbf{x}^{(t)} = \{\mathbf{v}_1^{(t)}, \dots, \mathbf{v}_\eta^{(t)}\}$  denote the swarm configuration and  $\mathcal{N}^{(t)}$  be the neighborhood system at a time step  $t$ . Under parallel Gibbs sampling, all robots will simultaneously update their velocities based on the configuration  $\mathbf{x}^{(t)}$ ; in particular, the  $i$ -th robot updates its instantaneous velocity  $\mathbf{v}_i^{(t)}$  to  $\mathbf{v}_i^{(t+1)}$  given the swarm configuration  $\mathbf{x}^{(t)}$  with probability

$$P_i(\mathbf{v}_i^{(t+1)} \mid \mathbf{x}_{\mathcal{R}\setminus i}^{(t)}, \mathbf{v}_i^{(t)}) = \frac{e^{-H(\mathbf{v}_i^{(t+1)}, \mathbf{x}_{\mathcal{R}\setminus i}^{(t)})}}{\int_{\Lambda_i} e^{-H(\mathbf{z}, \mathbf{x}_{\mathcal{R}\setminus i}^{(t)})} d\mathbf{z}}. \quad (4.3)$$

The conditional probability function in Equation (4.3) states that the likelihood of the  $i$ -th robot adopting a velocity  $\mathbf{v}_i^{(t+1)}$  is determined by weighing the immediate potential energy associated with this velocity against the potential energy corresponding to any feasible velocity  $\mathbf{z}_i \in \Lambda_i$  the robot could achieve, given the current instantaneous velocities of all other robots  $\mathbf{x}_{\mathcal{R}\setminus i}^{(t)}$ . That is, the robot still depends on the global knowledge to compute the potential energy  $H(\cdot, \mathbf{x}_{\mathcal{R}\setminus i})$ . However, if we express it in an extended form

$$H(\cdot, \mathbf{x}_{\mathcal{R}\setminus i}) = \left( U_{\{i\}}(\cdot) + \sum_{\forall j \in \mathcal{R}\setminus i} U_{\{j\}}(\mathbf{v}_j) \right) + \sum_{\forall j \in \mathcal{N}_i} U_{\{i,j\}}(\cdot, \mathbf{v}_j), \quad (4.4)$$

one may note that the second term inside the parenthesis is constant for the  $i$ -th robot, which lets us reduce the form for  $H(\mathbf{v}_i^{(t+1)}, \mathbf{x}_{\mathcal{R}\setminus i}^{(t)})$  and  $H(\mathbf{z}_i, \mathbf{x}_{\mathcal{R}\setminus i}^{(t)})$ . Especially, for the  $i$ -th robot, its potential energy can be written as,

$$H_i(\mathbf{x}_{\mathcal{R}\setminus i}) = U_{\{i\}}(\mathbf{v}_i) + \sum_{\forall j \in \mathcal{N}_i} U_{\{i,j\}}(\mathbf{v}_i, \mathbf{v}_j). \quad (4.5)$$

This shows the local nature of the Gibbs potential energy and implies that we do not require the knowledge of the entire swarm to sample velocities for the  $i$ -th robot, but only information about its neighbors  $\mathcal{N}_i^{(t)}$ . Thus, we can rewrite (4.3) as

$$P_i(\mathbf{v}_i^{(t+1)} | \mathcal{N}_i^{(t)}, \mathbf{v}_i^{(t)}) = \frac{e^{-\left( U_{\{i\}}(\mathbf{v}_i^{(t+1)}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)}) \right)}}{\int_{\Lambda_i} e^{-\left( U_{\{i\}}(\mathbf{z}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{z}, \mathbf{v}_j^{(t)}) \right)} d\mathbf{z}}. \quad (4.6)$$

Note that directly evaluating the conditional probability function in Equation (4.6) is challenging due to the high cardinality of the configuration space, which makes the computation of the normalizing constant intractable in practice. One approach to obtaining the most likely value is to use a maximum likelihood estimation method. This involves taking the derivative of the probability function with respect to the variable, setting it equal to 0, and solving for the value that maximizes the probability. However, this is impossible since the derivative of equation (4.6) is not defined.

Alternatively, if it is not possible to directly evaluate the probability function, one can use a Monte Carlo method to approximate the maximum likelihood value. This involves sampling values from the probability function using a sampling algorithm and then selecting the most frequently occurring value as the maximum likelihood estimate. In the following, we describe a method for sampling the maximum likelihood velocities using the probability (4.6).

## 4.4 Sampling Algorithm

The Metropolis-Hastings algorithm (Hastings, 1970) is a widely employed method for sampling from a probability distribution, particularly when the normalizing constant is unknown. It is a Markov Chain Monte Carlo (MCMC) algorithm, which means that it generates samples by constructing a Markov chain that has the desired probability distribution as its stationary distribution.

In this specific application within swarm robotics, the Metropolis-Hastings algorithm iteratively estimates the maximum likelihood velocities for the swarm members.

---

**Algorithm 1** Sampling most likely velocity command  $\mathbf{v}_i^{(t+1)}$  for  $i$ -th robot

---

```

1: procedure UPDATE( $\mathbf{v}_i^{(t)}, \mathcal{N}_i^{(t)}$ ) ▷ Current state
2:    $\mathbf{V}^{(0)} \leftarrow \mathbf{v}_i^{(t)}$ , ▷ Set of velocities
3:    $\mathbf{U}^{(0)} \leftarrow H(\mathcal{N}_i^{(t)}, \mathbf{v}_i^{(t)})$ , ▷ Set of potential energies
4:   for  $k \leftarrow 1$  to  $\mathcal{I}$  do
5:      $\mathbf{v}_i^{(t+1)} \leftarrow N(\mathbf{v}_i^{(t)}, \boldsymbol{\Sigma})$ , ▷ Gaussian sampling
6:      $\mathbf{u} \leftarrow H(\mathcal{N}_i^{(t)}, \mathbf{v}_i^{(t+1)})$ ,
7:      $\Delta E \leftarrow \mathbf{u} - \mathbf{U}^{(k-1)}$ , ▷ Potential energy variation
8:      $g \leftarrow \exp(-\Delta E)$ , ▷ Compute the Gibbs energy
9:      $r \leftarrow \mathcal{U}(0, 1)$ , ▷ Uniform sampling
10:    if  $(\Delta E < 0) \vee (r < g)$  then ▷ Sample accepted
11:       $\mathbf{V}^{(k)} \leftarrow \mathbf{v}_i^{(t+1)}$ ,
12:       $\mathbf{U}^{(k)} \leftarrow \mathbf{u}$ ,
13:    else ▷ Sample rejected
14:       $\mathbf{V}^{(k)} \leftarrow \mathbf{V}^{(k-1)}$ ,
15:       $\mathbf{U}^{(k)} \leftarrow \mathbf{U}^{(k-1)}$ ,
16:     $\mathbf{V} \leftarrow \mathbf{V}^{(j, \dots, \mathcal{I})}$ , ▷ Reject the first  $j$  velocities
17:     $\mathbf{v}_i^{(t+1)} \leftarrow (\mathbf{V}^q + \dots + \mathbf{V}^{\mathcal{I}}) / (\mathcal{I} - q)$ , ▷ Average
18:    return  $\mathbf{v}_i^{(t+1)}$ 

```

---

The algorithm considers the current instantaneous velocity and the relative positions of neighbors. Throughout the iterations, this algorithm explores the space of possible velocities, considering the swarm's local interactions and constraints. The set of candidate velocities is then employed to compute the maximum likelihood velocity, achieved by taking the candidates' average. This approach ensures that the sampled velocities align with the underlying probability distribution, providing a valuable tool for optimizing the swarm's behavior and adaptability in dynamic environments. The modified Metropolis-Hastings algorithm for this specific context is presented in Algorithm 1, enabling the efficient sampling of maximum likelihood velocities based on the probability (4.6).

## 4.5 Potential Energy

When designing a potential energy function in the context of swarm robotics, it is common to consider two potential functions that promote behaviors in the swarm. For

example, one potential function may promote aggregation or dispersion of the robots while another promotes cohesive navigation behavior. Combining these two potential functions into a single potential energy function makes it possible to achieve different behaviors in the swarm. This can be done by simply adding the two potential functions together, resulting in a new potential function of the form  $U(\cdot) = U_1(\cdot) + U_2(\cdot)$ , where  $U_1(\cdot)$  and  $U_2(\cdot)$  are pairwise interaction potential functions promoting aggregation and cohesive navigation, respectively.

The specific form of the potential functions will depend on the specific details of the problem at hand, but we propose in this dissertation the use of two potential functions that seem to effectively encode a wide range of swarm behaviors.

### 4.5.1 Coulomb-Buckingham Potential

The Coulomb-Buckingham potential (Buckingham, 1938) is a combination of the Lennard-Jones potential and the Coulomb potential, which are commonly used to describe the interactions between particles. The Lennard-Jones potential is a mathematical function that describes the attractive and repulsive forces between neutral atoms or molecules, while the Coulomb potential is used to describe the interactions between charged particles.

The Coulomb-Buckingham potential is given by the following formula:

$$\Phi(r) = \varepsilon \left( \frac{6}{\alpha - 6} e^{\alpha} \left( 1 - \frac{r}{r_0} \right) - \frac{\alpha}{\alpha - 6} \left( \frac{r_0}{r} \right)^6 \right) + \frac{c_i c_j}{4\pi \varepsilon_0 r}, \quad (4.7)$$

where  $r = \|\mathbf{p}_{ij}\|$  is the euclidean distance between the particles  $i$  and  $j$ ;  $\varepsilon$  is the depth of the minimum energy;  $r_0$  is the minimum energy distance;  $\alpha$  is a free dimensionless parameter;  $c_i$  and  $c_j$  are the charges of the particles  $i$  and  $j$ ; and  $\varepsilon_0$  is an electric constant.

In this dissertation, we took advantage of the Coulomb-Buckingham potential to model group formation and the heterogeneity of the swarm by setting the particle charges. This is an interesting technique when using the Coulomb-Buckingham potential, as it allows one to control the strength and range of the interactions between the particles based on their charges. By setting the charges of the particles appropriately, one can use

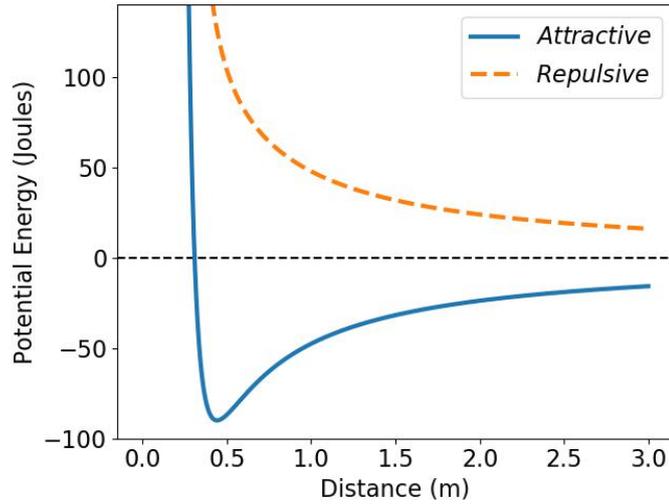


Figure 4.3: The Coulomb-Buckingham potential function. It depends on the distance  $r$  among the  $i$ -th and  $j$ -th robots and a function  $C(i, j)$  that produces attractive or repulsive behaviors depending on the heterogeneity among them. Here we assume  $|c_i c_j| = 24$ ;  $\varepsilon = \varepsilon_0 = 0.04$ ;  $r_0 = 1.6$ ; and  $\alpha = 1.0$ .

the Coulomb-Buckingham potential to model a wide range of swarm behaviors, such as aggregation, dispersion, collision avoidance, and collective decision-making.

From (4.7), one may set the interaction among the  $i$ -th and  $j$ -th robots by replacing the product  $c_i c_j$  by the following function,

$$C(i, j) = (2 \mathbb{1}((i, j) \in \tau_k) - 1) |c_i c_j|, \quad (4.8)$$

where  $\mathbb{1}(\cdot)$  denotes the indicator function. In this way  $C(i, j)$  will be positive if the  $i$ -th and  $j$ -th robots belong to the same group  $\tau_k$  and negative otherwise. Figure 4.3 illustrates the Coulomb-Buckingham potential encoding attractive and repulsive behaviors.

## 4.5.2 Kinetic Energy

In classical mechanics, the kinetic energy of a system of particles can be computed using the following formula:

$$E_k = \frac{1}{2} m \mathbf{v}^2, \quad (4.9)$$

where  $m$  is the mass of the particle and  $\mathbf{v}$  is its velocity.

In this dissertation, we are interested in computing the kinetic energy produced by the relative velocities of the  $i$ -th robot's neighbors. To do this, one may first need to define the magnitude resultant of the relative velocities among all of the  $i$ -th robot's neighbors that are of the same type as,

$$\mathbf{V}_i = \sum_{\forall j \in \mathcal{N}_i \wedge (i,j) \in \tau_k} \mathbf{v}_{ij}. \quad (4.10)$$

Note that by computing the  $\mathbf{V}_i$  using relative velocities, only when  $\mathbf{V}_i \rightarrow 0$ , there is a duality in the behavior produced by the robots. More specifically, one may not differentiate if the robots are stationary or moving at the same velocities. To avoid such duality, we added a second term to force the  $i$ -th robot to reach its maximum speed,

$$\mathbf{V}_i = \sum_{\forall j \in \mathcal{N}_i \wedge (i,j) \in \tau_k} \mathbf{v}_{ij} + (v_{max} - \mathbf{v}_i). \quad (4.11)$$

Then, the kinetic energy relative to the  $i$ -th robot can be computed as:

$$\mathbf{E}_k(\mathbf{V}_i) = \frac{1}{2}m(\mathbf{V}_i \cdot \mathbf{V}_i), \quad (4.12)$$

where  $m$  is the cumulative mass of the group, that is

$$m = m_i + \sum_{\forall j \in \mathcal{N}_i \wedge (i,j) \in \tau_k} m_j, \quad (4.13)$$

and  $(m_i, m_j)$  may be taken as a heterogeneity factor.

## 4.6 Designing Swarm Behaviors

This dissertation demonstrates that the appropriate engineering of potential energy, achieved through the strategic combination of previously introduced potential functions, may yield diverse behaviors within a swarm. Practically, this potential energy translates the current swarm configuration into a single energy value. Through the conditional probability function outlined in Equation (4.6), one may sample instantaneous

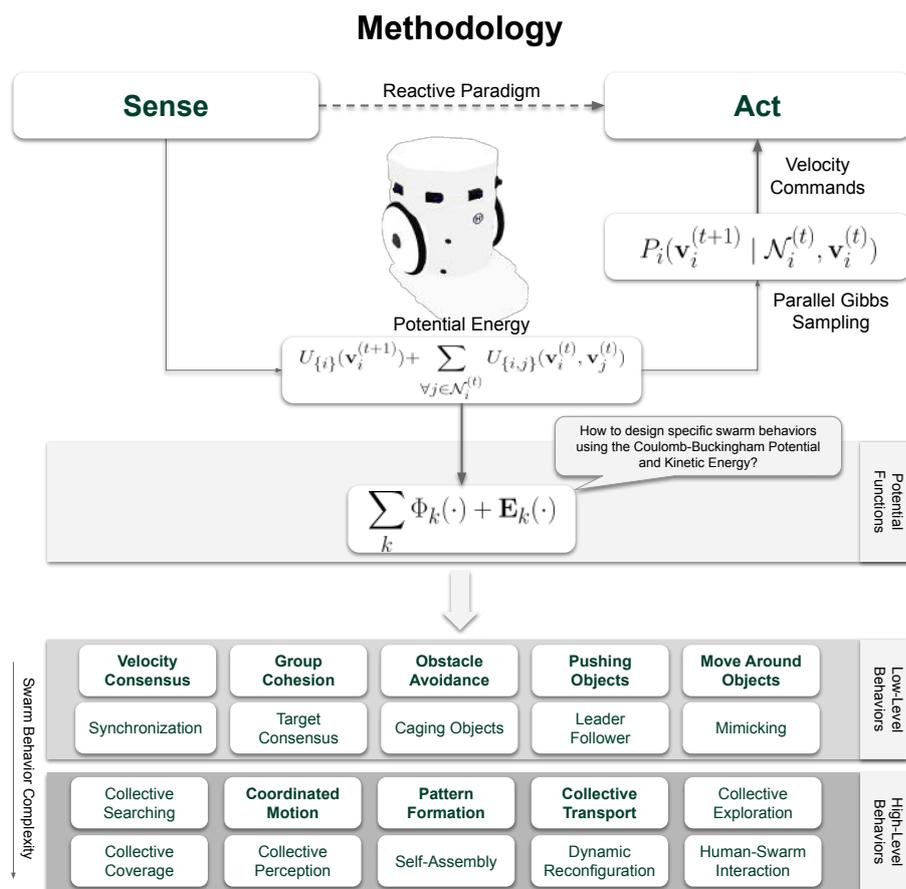


Figure 4.4: Methodology for generating diverse swarm behaviors through Coulomb-Buckingham potential and kinetic energy combinations in various configurations. The highlighted behaviors are tackled in this dissertation.

velocities that globally minimize this energy value over time. The minimum energy value represents swarm configurations that display the desired behaviors. However, establishing a clear and direct correlation between the potential energy function and the ensuing swarm behavior can be challenging, especially for intricate behaviors. It demands a blend of intuition and creative insight to design the potential energy function in a manner that produces the desired behavior. To provide further clarity, an overview of our methodology is depicted in Figure 4.4. Additionally, we establish a guideline outlining the sequential steps of our approach, aiding in a comprehensive understanding.

1. **Potential Functions as Primitives:** Potential functions, such as the Coulomb-Buckingham potential and kinetic energy, serve as mathematical descriptions of primitive behaviors. The Coulomb-Buckingham potential describes the electrostatic interaction between charged particles and may be used as an attraction or repulsion

component to get more intricate behaviors. When combined with the kinetic energy, a component for motion interactions may influence the swarm behavior dynamics.

2. **Model Low-Level Swarm Behaviors:** The primitive behaviors (potential functions) can be combined in various forms to encode more intricate swarm behaviors, such as group formation, cohesive navigation, and robot-environment interaction. This stage demands a blend of intuition, creative thinking, and tests to engineer a potential energy function that yields the intended behaviors. These low-level behaviors serve as building blocks for higher-level swarm behaviors.
3. **Model High-Level Swarm Behaviors:** Use these low-level swarm behaviors to design high-level potential functions. These functions account for more complex swarm behaviors like collective transport, exploration, coverage, searching, and more. Particularly for highly complex swarm behaviors, it may be necessary to dynamically adjust potential functions based on the evolving swarm configuration.
4. **Parallel Gibbs Sampling:** Refer to the probability function defined in Equation (4.6) which is computationally executed through Algorithm 1. Understand that it describes the likelihood of a specific swarm configuration based on the designed potential energy. By designing the potential energy function in an appropriate way, it is possible to control the stationary distributions of this probability. Consequently, sampling over this probability serves as a control mechanism that gives maximum likelihood command velocities. By following these commands, eventually, the swarm reaches configurations that minimize the global potential energy while emerging with specific scenarios.
5. **Iterate and Experiment:** Finally, embrace an iterative approach. Test and refine potential energy functions to fine-tune swarm behaviors. Experimentation and analysis may help optimize the designed behaviors for specific applications.

The following three chapters of this dissertation describe how the methodology has been applied to produce three intricately designed swarm behaviors: flocking and segregation, cooperative object transportation, and pattern formation.

# Chapter 5

## Flocking-Segregative Behavior

This chapter introduces an application of our methodology to model decentralized interactions among robots in a heterogeneous swarm, relying solely on local sensing. Through careful design of the potential energy function, it is possible to simultaneously induce segregative and flocking behaviors within the swarm.

### 5.1 Introduction

Among the fundamental behaviors required by a robotic swarm, group formation and cohesive navigation stand as paramount (Brambilla et al., 2013). Within this context, segregation emerges as a specialized form of group formation, involving the clustering of robots with shared characteristics while maintaining separation from other groups (Santos et al., 2014a). Additionally, cohesive navigation plays an essential role in ensuring the safe and efficient movement of a group of robots. Flocking behavior exemplifies one method of achieving cohesive navigation, wherein individuals synchronize their velocities so they move together as a unified group (Reynolds, 1987). While classical models for flocking support the aggregation of robots, to our knowledge, there exists no prior literature addressing simultaneous flocking and segregation from a randomly initialized state, especially within the context of multiple distinct groups relying solely on local sensing.

These behaviors find applications in various domains, including area coverage, surveillance and reconnaissance, transport, foraging, among others. For instance, a group



Figure 5.1: Flocking-segregative behavior. A swarm of heterogeneous robots demonstrates the ability to simultaneously exhibit segregation and flocking behaviors, relying solely on local sensing. The robots navigate a physically simulated environment containing obstacles, autonomously forming distinct groups while maintaining cohesive navigation through the terrain. This illustrates the concurrent manifestation of both segregation and flocking behaviors.

of heterogeneous robots engaged in a self-assembly task can utilize segregation to form specific groups and subsequently navigate each part to construct sophisticated structures. This chapter introduces a novel stochastic and decentralized method that enables a swarm of heterogeneous robots to achieve simultaneous segregation and flocking behaviors using only local sensing, as illustrated in Figure 5.1. The approach leverages the concepts explained in Chapter 4 to model the robot swarm as a Gibbs Random Field (GRF) and design an appropriate potential energy function that yields such behaviors.

As a consequence, in addition to supporting the segregation and navigation of different groups while avoiding collision with obstacles, the approach allows the swarm to reach configurations sufficiently close to the global minimum energy. Through simulated experiments, we contrast our method with a deterministic gradient descent-type algorithm using potential differentials, demonstrating that such a mechanism can be easily trapped at local minima of potential. Additionally, we compare our segregative behavior with some of the state-of-the-art approaches and evaluate the flocking behavior in the presence of noise. Further, physical simulations exhibit remarkable resilience in executing these behaviors even within complex environments containing obstacles or in cases where robots fail and stop moving. Finally, we conducted real-world experiments, serving as a compelling proof-of-concept for the efficacy of our approach.

## 5.2 Related Work

Most works in swarm robotics traditionally centered on homogeneous systems, characterized by robots possessing identical attributes, capacities, and functionalities (Dudek et al., 1996). However, recent years have witnessed a growing interest in heterogeneous systems, marked by robots exhibiting diverse characteristics. As a consequence, this has led to a thorough exploration of emergent behaviors within such mixed collective paradigms, notably the phenomenon of segregation. In this context, Groß et al. (2009) introduced a seminal control algorithm, drawing inspiration from the Brazil Nut effect – a collective phenomenon wherein segregation manifests through the sustained agitation of a particle mixture comprising distinct sizes. Subsequent advancements were made by Chen et al. (2012) and Joshi et al. (2019), where they conducted performance evaluations and refinements of the algorithm and presented experiments with real robots.

Another approach to segregate a swarm of heterogeneous robots was presented by Kumar et al. (2010). The authors took inspiration from a biological theory that explains how differences in cell adhesion generate mechanical forces that drive cellular segregation (Differential Adhesion Hypothesis) (Steinberg, 1963). This mechanism is modeled with the concept of differential potential, in which robots are subjected to differential artificial potential fields according to their groups. The method’s convergence is guaranteed for two classes, but the swarm may be trapped in local minima when more classes are employed. This approach was later extended in Santos et al. (2014a, 2020) to deal with more than two groups of robots. One limitation is the requirement that robots must have global information about the positions of other robots.

Motivated by the use of differential artificial potential fields, Edson Filho and Pimenta (2015) proposed a novel controller that differs from the previous ones by using abstractions (Belta and Kumar, 2004) to represent each group. One advantage of such a controller is that it may not require that all robots receive information from all other robots all the time. More recently, the authors extended this controller to incorporate a collision avoidance scheme that does not interfere with the original segregation

controller (Ferreira-Filho and Pimenta, 2019). In a different work, they presented a decentralized control strategy to segregate heterogeneous robot swarms distributed in curves using consensus protocols and heuristics to compute the traveled geodesic distances on curves (Ferreira Filho and Pimenta, 2020). This approach assumes that robots know the curve and maintain an underlying fixed communication topology.

Recently, two works assuming minimal and local-only requirements for segregating a swarm of heterogeneous robots have been proposed. Mitrano et al. (2019) extended the concept of a minimalistic reactive controller (St-Onge et al., 2018) to achieve segregation. They demonstrate that robots with only a ternary sensor and a controller that maps sensor readings to wheel speeds can reach a segregated state. Additionally, considering local sensing and a memory mechanism supported by communication, Inácio et al. (2019) proposed a strategy that combines concepts of Particle Swarm Optimization (PSO) (Eberhart and Kennedy, 1995) with the Optimal Reciprocal Collision Avoidance (ORCA) (van den Berg et al., 2011) to archive segregation.

Following the works that rely only on local information, we assume that our robots only use their neighbors' distances and relative velocities to achieve simultaneous segregation and cohesive navigation. However, most of these works only consider collisions with other robots. Our method also allows the robots to avoid collisions with obstacles in the environment, considering that they are equipped with sensors capable of differentiating between obstacles and robots. Besides dealing with the segregation problem, our approach also generates a cohesive navigation behavior for the different groups of robots. One of the main challenges in achieving such behavior is reaching a consensus on each part of the group's velocity as its size increases. In addition to that, the groups must remain segregated while navigating.

One of the earliest and most influential approaches to steer a swarm of homogeneous agents using only local interactions was proposed by Reynolds (1987). This mechanism, called *boids*, combines three simple rules: separation, cohesion, and alignment. While most works on this subject deal with homogeneous groups, some use heterogeneous robotic swarms to study the flocking of distinct groups. Momen et al. (2007) extended the

flocking mechanism with heterospecific attraction rules (Mönkkönen et al., 1990) to model different attraction forces between two groups of robots, producing a mixed-species flocking. Ducatelle et al. (2011) proposed a mechanism that emerges cooperative self-organized behaviors to solve complex tasks using simple local interactions between the robots of the two different groups. Another study on self-organized flocking explored the concept of Swarm heterogeneity in the sense that robots with more capabilities help others that lack some capabilities in order to yield the desired behavior (Stranieri et al., 2011).

Some works have addressed the challenge of maintaining segregation during navigation, where robots initiate from a segregated state and must maintain this separation throughout their movement. For instance, Santos et al. (2014b) introduced the concept of Virtual Group Velocity Obstacles, combining principles from both flocking (Reynolds, 1987) and Velocity Obstacles (Fiorini and Shiller, 1998) with abstract representations of the groups. In an effort to enhance performance beyond this approach, Inácio et al. (2018) proposed a combination of the Optimal Reciprocal Collision Avoidance algorithm (Van Den Berg et al., 2011) with flocking concepts.

The method proposed in this chapter stands out from previous works by simultaneously inducing segregation and flocking behaviors. Starting from an entirely random configuration, the robots autonomously self-organize into distinct groups while maintaining this separation throughout their navigation. As far as our knowledge extends, this research marks the pioneering introduction of a fully decentralized stochastic controller that proficiently produces both behaviors, exclusively relying on local interactions.

Furthermore, it is worth noting that although our approach achieves flocking behaviors, we do not rely on or extend the mechanism originally proposed by Reynolds (1987). Instead, we adopt the steps proposed in Chapter 4 to design a control mechanism that yields such behavior and combines it with the segregation paradigm. In the subsequent section, we delve into a detailed description of our method, explaining the intricacies of the potential energy function designed to produce these behaviors.

## 5.3 Method

Applying the concepts described in Chapter 4, we design an effective potential energy function that combines the Coulomb-Buckingham potential and kinetic energy in a way that implicitly induces a swarm of heterogeneous robots to achieve segregation and flocking behaviors simultaneously while avoiding obstacles in the environment. The underlying idea is that the Coulomb-Buckingham potential enables robots of the same group to aggregate together and robots of other types to repel or also to avoid obstacles in the environment. The kinetic energy helps the robots to consense their relative velocity concerning their neighborhood and hence to harmony navigate the group through the environment. A formalization of our method is given in the following.

### 5.3.1 Formalization

Initially, let us assume the conditional probability function in Equation (4.6),

$$P_i(\mathbf{v}_i^{(t+1)} | \mathcal{N}_i^{(t)}, \mathbf{v}_i^{(t)}) = \frac{e^{-\left( U_{\{i\}}(\mathbf{v}_i^{(t+1)}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)}) \right)}}{\int_{\Lambda_i} e^{-\left( U_{\{i\}}(\mathbf{z}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{z}, \mathbf{v}_j^{(t)}) \right)} d\mathbf{z}}.$$

where we state that by properly setting the potential energy (exponential term), it is possible to control the stationary distributions of this probability and by sampling most likely instantaneous velocity  $\mathbf{v}_i^{(t+1)}$ , eventually, the swarm produces behaviors in a decentralized manner, using only local information.

The potential energy in Equation (4.6) given the perception of neighborhood  $\mathcal{N}_i^{(t)}$  at time step  $t$  and a arbitrary velocity  $\mathbf{v}_i^{(t+1)}$ ,

$$H_i(\mathbf{v}_i^{(t+1)}, \mathcal{N}_i^{(t)}) = U_{\{i\}}(\mathbf{v}_i^{(t+1)}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)}), \quad (5.1)$$

reflects the locality constraint where the  $i$ -th robot only has access to its local state, and it combines two potential functions that are convenient to encode different swarm behaviors. The first potential function term depends exclusively on the state of the  $i$ -th robot. It may be interesting in representing individual behaviors, while the second term takes account of the neighborhood state and may encode collective behaviors. The following are our designs for each of these potential functions to induce segregation and flocking behaviors.

First, let us design the individual potential  $U_{\{i\}}$  to induce an obstacle avoidance behavior. Practically, the underlying idea is to transform the perception of the robot regarding obstacles in its surroundings into a quantifiable potential energy value. When the robot is close to obstacles, this value should be a high positive, indicating that configuration is not likely for the conditional probability in Equation (4.6). Conversely, if no obstacles are in close range, the value should approach zero or be exactly zero. Assuming the robot can estimate its distance to points on obstacles (as outlined in Definition 4.4), we can employ the Coulomb-Buckingham potential to translate this information as a repulsive behavior. Formally, we define the obstacle avoidance behavior as follows:

$$U_{\{i\}}(\mathbf{v}_i^{(t+1)}, \mathcal{W}_i^{(t)}) = \sum_{\forall j \in \mathcal{W}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{w}_j^{(t)}\|). \quad (5.2)$$

Here,  $\mathcal{W}_i^{(t)}$  contains the points detected along the perimeters of obstacles at time step  $t$ . The term  $\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{w}_j^{(t)}\|$  represents the Euclidean distance between the predicted relative position  $\mathbf{p}_i^{(t+1)}$  of the  $i$ -th robot, calculated using the motion model  $\mathcal{K}$  as described in Definition 4.2, assuming the velocity  $\mathbf{v}_i^{(t+1)}$ , and the  $j$ -th point on the obstacle. The heterogeneity function  $C(i, j)$  used by Coulomb-Buckingham potential,  $\Phi(\cdot)$ , is set strictly positive, inducing a repulsion force. In essence, this potential function evaluates the energy value based on the robot configuration at time step  $t$ , assuming the velocity  $\mathbf{v}_i^{(t+1)}$ .

Now that we have established a collision avoidance behavior, we turn our attention to designing the second-term potential  $U_{\{i,j\}}$  in Equation 5.1 to induce segregative and flocking behaviors. To account for the segregative behavior, it is imperative that robots maintain proximity to others of the same type while keeping a distance from those of different types. Thus, we require a potential function that translates neighborhood information into an energy value summarizing the segregation state. Once again, the

Coulomb-Buckingham potential, with the heterogeneity function  $C(i, j)$ , can be effectively employed to model such behavior by selectively applying attraction and repulsive forces. In addition to segregation, we aim to induce flocking behavior, wherein robots collectively align their velocities and move as a cohesive group through the environment. One effective strategy to implicitly represent the relative velocities of the group as an energy value is by computing their kinetic energy. In a consensus state of motion, the relative velocity tends to be close to zero, indicating an optimal configuration. Formally, we presented the potential function that induces flocking and segregation behavior as

$$\sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j) = \sum_{\forall j \in \mathcal{N}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathcal{K}(\mathbf{p}_j^{(t)}, \mathbf{v}_j^{(t)})\|) + \mathbf{E}_k(V_i^{(t)}), \quad (5.3)$$

where the first term defines the attraction and repulsion between  $i$ -th and  $j$ -th robots given their heterogeneity and the second term accounts for consensus on the relative velocity of the neighbors. The potential is computed for the state of the  $i$ -th robot when it takes the velocity  $\mathbf{v}_i^{(t+1)}$  and the  $j$ -th robots when it keeps the current measured velocity  $\mathbf{v}_j$ . In essence, we use the robot motion model to compute the predictive potential energy when the robot takes a velocity  $\mathbf{v}_i^{(t+1)}$ . Furthermore, it is important to notice that parameters for both potential functions, such as mass  $m$  and electrical charge  $c$  of the robots, should be properly adjusted in order to reach flocking segregative behaviors simultaneously. If the mass of the robot is excessively high, the flocking behavior should be more prominent. Conversely, an elevated electrical charge leads the robot to emerge with segregation but hinders the consensus of velocities.

Finally, through the combination of the potential functions (5.2) and (5.3) into the potential energy outlined in Equation (5.1),

$$H_i(\mathbf{v}_i^{(t+1)}, \mathcal{N}_i^{(t)}, \mathcal{W}_i^{(t)}) = \sum_{\forall j \in \mathcal{W}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{w}_j^{(t)}\|) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathcal{K}(\mathbf{p}_j^{(t)}, \mathbf{v}_j^{(t)})\|) + \mathbf{E}_k(\mathbf{V}_i^{(t)}), \quad (5.4)$$

it possible to compute the conditional probability function in Equation (4.6) by using Algorithm (1), described in Chapter 4. As a result, the algorithm will give the most likely velocity for the  $i$ -th robot given its sensing information, and eventually, the swarm will converge to a segregative-flocking behavior.

## 5.4 Experiments and Results

In this section, we evaluate our proposed method through a series of simulated experiments. Initially, we analyze the segregative behavior by systematically varying configurations and comparing the results with other methods from the literature. Following this, we assess the flocking behavior by examining velocity consensus and cohesion in the presence of sensor noise. Additionally, we demonstrate the robustness of our approach through physical simulations, showcasing its effectiveness even in complex environments with obstacles or in the event of robot failures. Finally, we conducted real-world experiments, serving as a compelling proof-of-concept for the efficacy of our approach. Figure 5.2 illustrates the simultaneous flocking and segregative behavior achieved by our method. For a more dynamic display, a video of these experiments is accessible on YouTube<sup>1</sup>, and the corresponding source code is available on GitHub<sup>2</sup>.

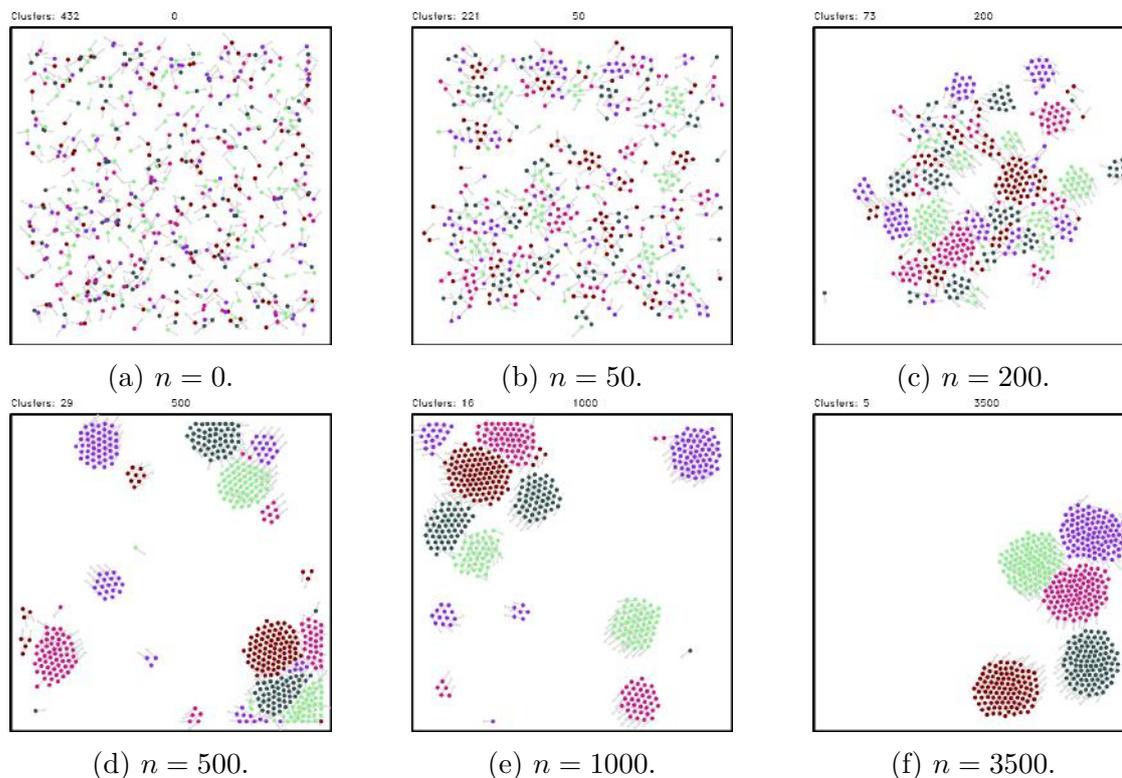


Figure 5.2: Illustration of segregative flocking involving five distinct groups, each comprising a hundred robots.

<sup>1</sup>Experiments video: <https://youtu.be/KooNGIStWlM>

<sup>2</sup>Source code: <https://github.com/verlab/2021-icra-grf-swarm>

### 5.4.1 Segregation Analysis

To evaluate the segregative behavior considering only local information, we conduct a comparative analysis of the convergence rates between our approach and those proposed by [Mitrano et al. \(2019\)](#) and [Inácio et al. \(2019\)](#). We consider the work proposed by [Santos et al. \(2020\)](#) as a baseline since it assumes global knowledge about the positions of other robots leading to a fast convergence rate. We also contrast our methodology with a deterministic gradient descent approach using potential differentials to show that such mechanism may be easily trapped at local minima.

The experiments consisted of 100 runs of each approach with a maximum of 20000 iterations. A random initial state is generated for each run, but it is the same for all approaches. At each iteration, the robot can move a maximum of 0.02 meters in a square area of 10 meters, with the walls as obstacles. We varied the number of robots and the number of heterogeneous groups to evaluate each approach's performance. As a metric, we compute the total amount of clusters formed by robots of the same type and the number of iterations necessary to reach it. Here, two robots of the same type are considered to be in the same cluster if their relative distance is less than 0.3 meters – the robot radius is 0.07 meters. The sensing range is set to  $\lambda = 0.5$  meters. Figure 5.3 shows the mean and the 99% confidence interval comparing our method with the other works.

Across various scenarios, our approach demonstrated competitive performance compared to the strategy presented by [Inácio et al. \(2019\)](#). Specifically, our method achieved superior segregation states, although the latter exhibited a faster convergence rate. Upon increasing the sensing range, we observed that our approach outperforms Inácio's method in terms of the number of iterations required, as the latter is more susceptible to congestion issues. The minimalist approach proposed by [Mitrano et al. \(2019\)](#) showcased promising results across various scenarios but generally is slower in convergence. Regarding the advantage of using a stochastic approach over a gradient descent-type algorithm, we observe that such a mechanism is easily trapped to a minimum of potential, preventing it from converging to better results in comparison to our method.

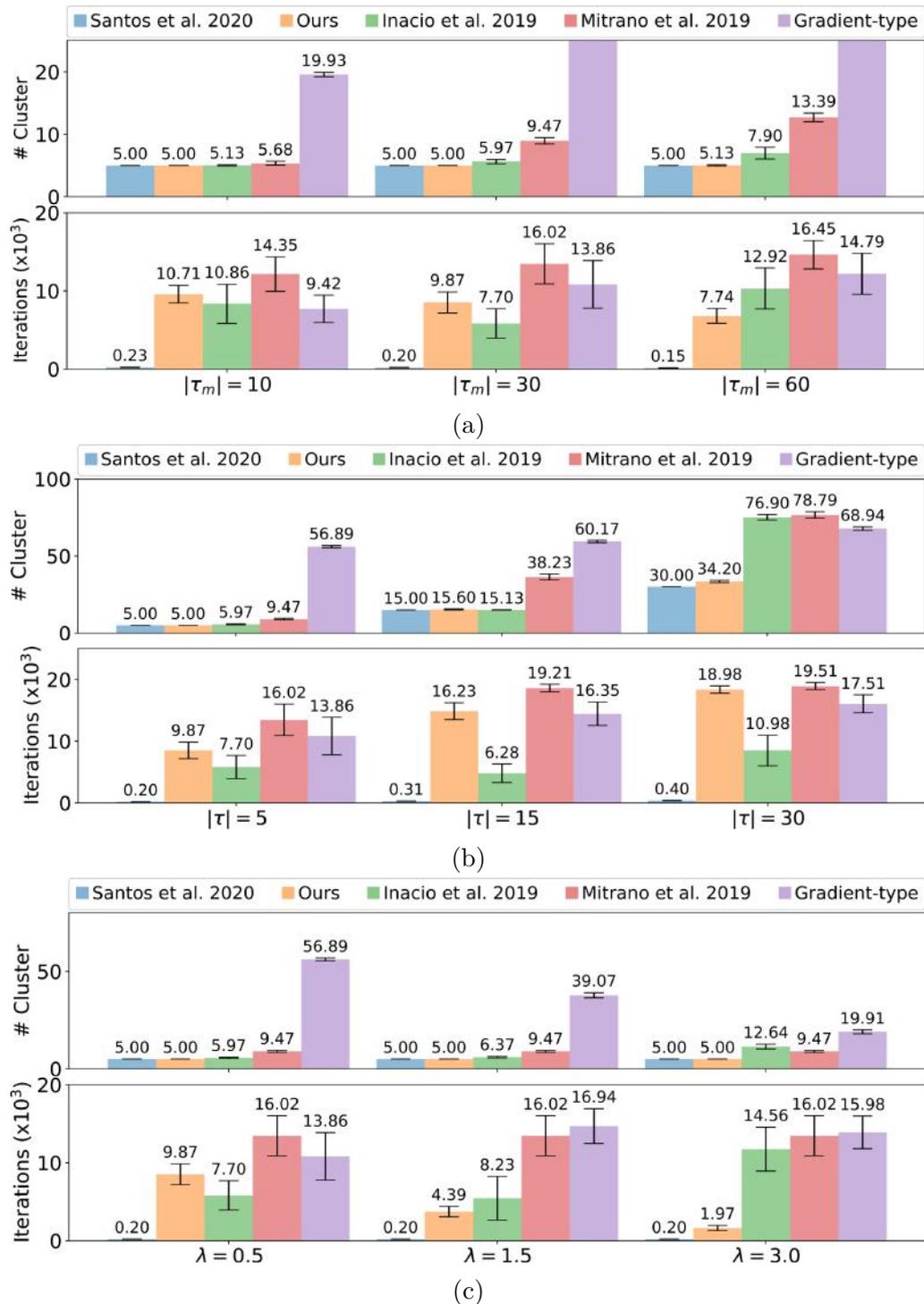


Figure 5.3: The minimum number of clusters yield by each approach in up to 20000 iterations when: (a) we increase the number of robots  $|\tau_m| = \{10, 30, 60\}$  keeping  $|\tau| = 5$  heterogeneous groups; and (b) we increase the number of groups  $|\tau| = \{5, 15, 30\}$  keeping  $|\tau_m| = 10$  robots per group; and (c) increases the sensing range  $\lambda = \{0.5, 1.5, 3.0\}$  keeping  $|\tau_m| = 30$  robots and  $|\tau| = 5$  heterogeneous groups.

Analyzing the segregation using the number of formed clusters, we can see that all approaches executed relatively well for a small number of groups, even for an increasing number of robots per group (Figure 5.3a, top). The exception is the Gradient-Descent method, which gets trapped in local minima and cannot reach a segregated state. When the number of groups increases, our approach significantly outperforms the others, with a performance close to the baseline, which uses global information (Figure 5.3b, top). When there is a large number of groups, robots usually get trapped by other groups and cannot reach a segregated state. By relying on the stochastic nature of the GRF, our approach can handle these situations better. When we increase the sensing range, our method continues to execute well while the others degrade slightly (Figure 5.3c, top). While our method takes advantage of a larger sensing area to “capture” more robots and converge faster, other methods, especially [Inácio et al. \(2019\)](#) that rely on communication among neighbors, can have problems reaching a consensus.

Regarding the performance in terms of the number of iterations to reach segregation, it is evident that the methods exhibit similar efficiency as the number of robots increases. However, it is noteworthy that all these methods display a substantial decrease in performance compared to the baseline, which uses global information (Figure 5.3a, bottom). An interesting observation emerges when varying the number of groups, indicating that [Inácio et al. \(2019\)](#) demonstrates better performance (Figure 5.3b, bottom). Nonetheless, a caveat arises, particularly for  $|\tau| = 30$ : our metric considers the number of iterations until reaching the minimum number of clusters. Inácio’s method, as previously discussed, often falls short of reaching the minimum number of clusters, converging faster but to a suboptimal configuration. In contrast, our method, although requiring more time due to its stochastic nature, boasts a considerably higher success rate. It’s important to note that our method, characterized by continuous movement, covers more distance than other approaches, which may be disadvantageous in scenarios with strict energy consumption constraints. However, this continuous motion enables our method to simultaneously produce flocking behavior, distinguishing it from other segregation methods, underscoring the versatility and dynamic capabilities of our approach in swarm robotics scenarios.

### 5.4.2 Flocking Analysis

To evaluate the effectiveness of our method in producing flocking behaviors, we carried out some experiments and analyzed them regarding the average distance (cohesion) and consensus velocity between robots of the same type. To assess the robustness of our method, we introduced Gaussian noise ( $\epsilon$ ) into the sensor model, thereby introducing uncertainty into distances and relative instantaneous velocity estimates. These experiments encompassed 100 runs, each capped at 20000 iterations. We configured the system with  $|\tau_m| = 30$  robots distributed across  $|\tau| = 5$  heterogeneous groups, and a sensing range ( $\lambda$ ) of 0.5 meters. The robots initiated each run in a randomly assigned initial state, executing both segregation and flocking behaviors within a 10 by 10 meter environment, with a maximum speed set to  $v_{max} = 1$  meter per second.

We systematically introduced noise levels of  $\epsilon = \{0\%, 2\%, 6\%, 10\%\}$  in the sensor information. For context, a noise level of  $\epsilon = 10\%$  translates to an error of up to  $10\% \cdot \lambda = 0.05$  meters in distance and  $10\% \cdot v_{max} = 0.10$  meters per second in speed. Figure 5.4 provides summarized results, presenting the mean and the 95% confidence interval, clearing light on the impact of noise on our method.

As expected, it is evident that the increment in noise levels within the sensor model substantially influences velocity consensus. Up until  $\epsilon = 6\%$ , the swarm successfully maintains the desired flocking-segregative behavior for the given experimental configuration. However, at  $\epsilon = 10\%$ , the degradation of velocity consensus becomes apparent, leading to the failure of the flocking behavior to converge. Despite this, even with the presence of noise, we consistently observed the ability to segregate the swarm into the minimal number of clusters, showcasing the robustness of our approach under these conditions.

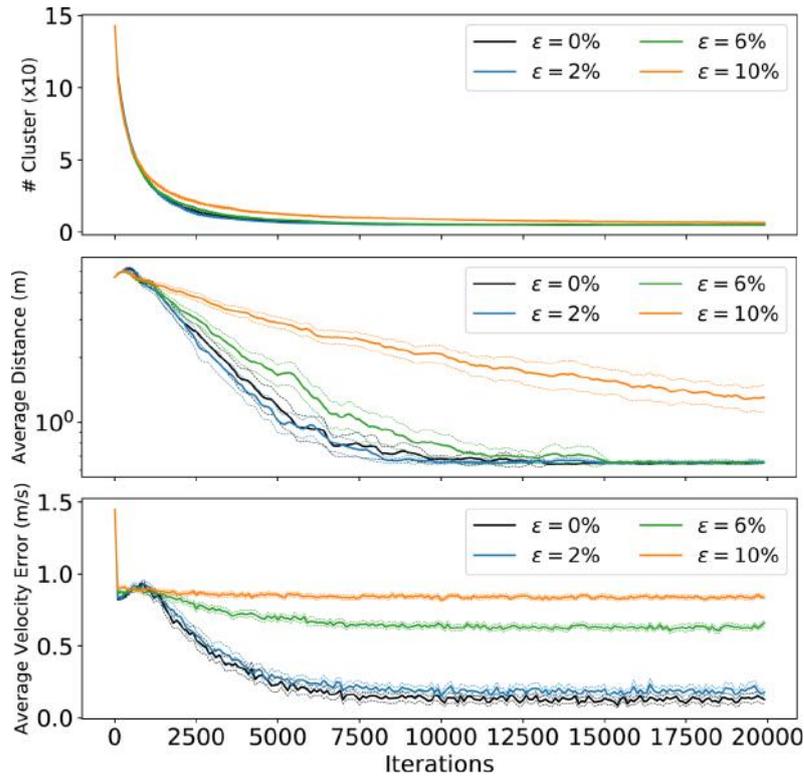


Figure 5.4: Impact of the noise in the sensor model on the performance of our method. The graphics display the number of clusters yield, the average distance, and velocity error among the same group of robots in up to 20000 iterations.

### 5.4.3 Flocking-segregative in Complex Environments

In this experiment, we delve into an evaluation of our method’s performance in achieving flocking-segregative behavior in complex environments. The implementation of our method has been seamlessly integrated with the Robot Operating System (ROS), allowing for decentralized execution, with each robot operating as an independent node or process. To simulate the intricate dynamics and interactions of the robots with their environment, we use the Gazebo simulator. Within this simulator, we designed five distinct environments namely from  $\mathcal{A}$  to  $\mathcal{E}$ , each characterized by a square layout spanning  $4 \times 4$  meters. These environments present a range of obstacle configurations, introducing diverse challenges that put our method to the test.

To assess the performance, we focus on segregation time to observe the impact of the environment on the method. In addition to varying the environment, we consider two distinct scenarios: one featuring two groups  $|\tau| = 2$ , each composed of  $|\tau_m| = 10$

robots, and the other comprising  $|\tau| = 4$  groups, each with  $|\tau_m| = 5$  robots. These configurations allow us to examine the scalability and adaptability of our method across varying group compositions. Here, the robots are bounded by a maximum velocity of 0.2 meters per second, while their sensing range extends up to  $\lambda = 0.3$  meters. By subjecting our method to these experiments, we display its efficacy and robustness in navigating complex environments, clearing light on its potential real-world applicability.

In the following, we evaluate the performance of our method in these environments. To ensure a thorough evaluation, we executed 15 trials for each configuration, recording and assessing the segregation time. The outcomes revealed interesting insights into the adaptability and resilience of our approach under varying environmental complexities.

In environment  $\mathcal{A}$ , depicted in Figure 5.5, the presence of only bordering walls as obstacles notably favored the performance of our method. This promoted a mean segregation time of  $50 \pm 20$  seconds for two groups and  $75 \pm 28$  seconds for four groups of robots, showcasing a baseline performance.

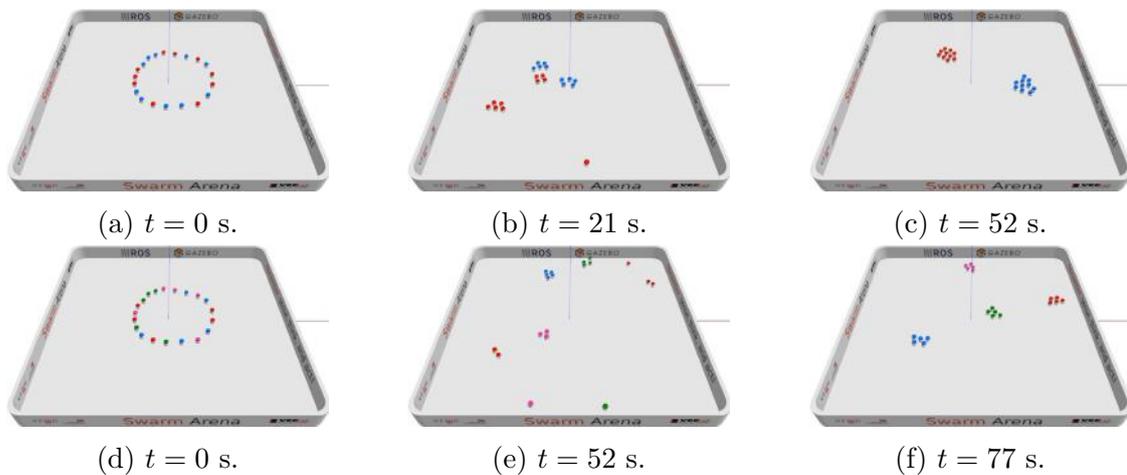


Figure 5.5: Flocking-segregative behavior demonstrated in simulated environment  $\mathcal{A}$ : (a)-(c) depict two groups of 10 robots each, while (d)-(f) showcase four groups of 5 robots demonstrating flocking segregation in a simplified environment. A video is available at <https://youtu.be/D9efvsX7wK4>.

In environment  $\mathcal{B}$ , depicted in Figure 5.6, we introduced an additional layer of complexity. In addition to bordering obstacles, a central corridor with two walls posed a slightly more challenging scenario. This environment impacts the mean convergence time to  $138 \pm 63$  seconds for two groups and  $278 \pm 67$  seconds for four groups of robots, demon-

strating our approach’s adaptability even in environments with increased complexity.

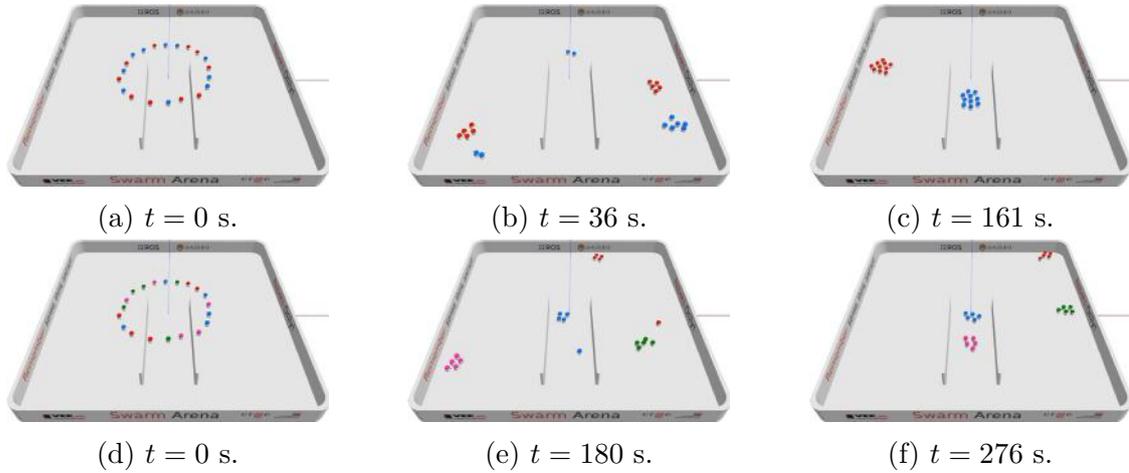


Figure 5.6: Flocking-segregative behavior demonstrated in simulated environment  $\mathcal{B}$ : (a)-(c) depict two groups of 10 robots each, while (d)-(f) showcase four groups of 5 robots demonstrating flocking segregation in an environment with walls forming a corridor. A video is available at <https://youtu.be/RQya4w01OMI>.

Environment  $\mathcal{C}$ , illustrated in Figure 5.7, presented a further increase in complexity. With internal walls partitioning the environment into rooms, this setup posed significant challenges by restricting robots from accessing each other. Despite this, our method demonstrated noteworthy adaptability reaching the segregation state in all cases. However, the mean convergence time of  $255 \pm 85$  seconds for two groups and  $618 \pm 135$  seconds for four groups of robots particularly increases compared with the other scenarios.

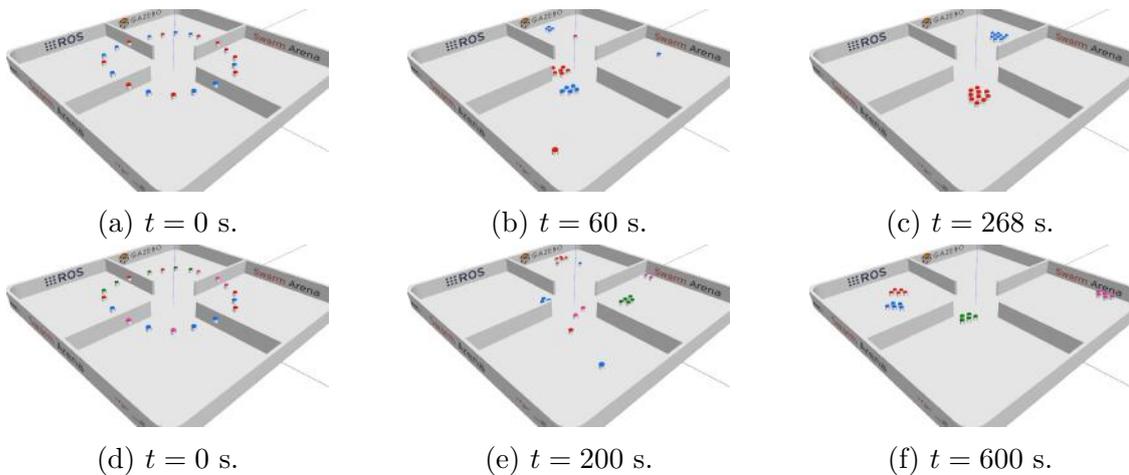


Figure 5.7: Flocking-segregative behavior demonstrated in simulated environment  $\mathcal{C}$ : (a)-(c) depict two groups of 10 robots each, while (d)-(f) showcase four groups of 5 robots demonstrating flocking segregation in a complex environment with walls forming four rooms. A video is available at <https://youtu.be/1wAA4mLDFLc>.

In environment  $\mathcal{D}$ , depicted in Figure 5.8, the introduction of walls forming a central 4-way corridor presented another challenging scenario. Arranging through these corridors, especially for a larger number of robots, introduced additional complexities that may impact the performance of our method. The environment induces a mean convergence time of  $130 \pm 52$  seconds for two groups and  $463 \pm 79$  seconds for four groups of robots, underscoring the methodology’s resilience in intricate settings.

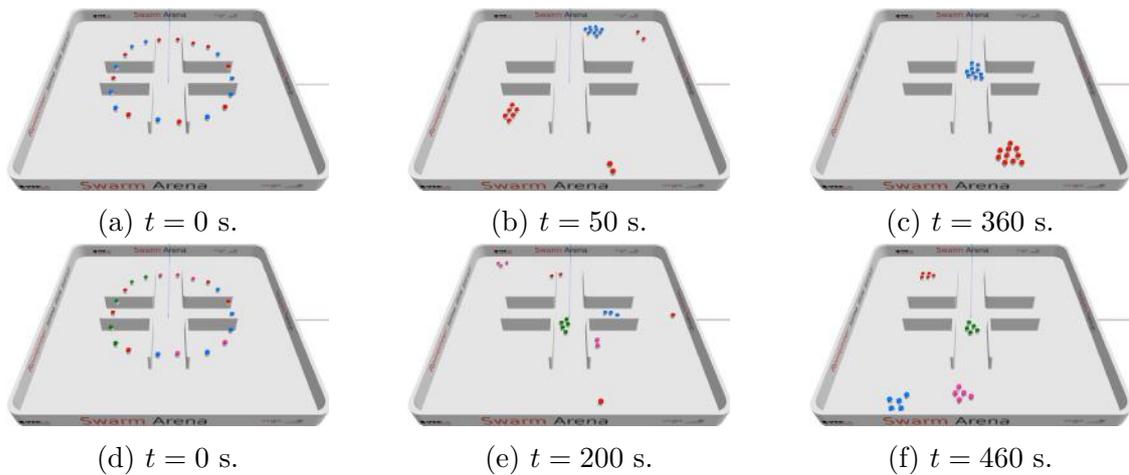


Figure 5.8: Flocking-segregative behavior demonstrated in simulated environment  $\mathcal{D}$ : (a)-(c) depict two groups of 10 robots each, while (d)-(f) showcase four groups of 5 robots demonstrating flocking segregation in a complex environment with walls forming four cross-shaped corridors. A video is available at <https://youtu.be/vakUTpRs1QE>.

Finally, in environment  $\mathcal{E}$ , as depicted in Figure 5.9, the inclusion of walls forming a room with a single restricted entrance heightened the challenges. This environment, with its potential to trap robots, presented a challenging scenario for our method. Nevertheless, it demonstrated impressive adaptability, yielding a mean convergence time of  $216 \pm 68$  seconds for two groups and  $372 \pm 69$  seconds for four groups of robots.

Summarizing the results regarding segregation time across all environments in Figure 5.10, we observed that environment  $\mathcal{C}$  emerged as the most challenging, especially when employing four groups, necessitating a longer convergence time compared to others. Additionally, we consistently observe that swarms with few heterogeneous groups perform faster than configurations with many groups, as demonstrated in the segregation analysis. Finally, even within these intricate environments, the robots exhibited commendable adaptability, eventually achieving the segregation state in all trials. Notably,

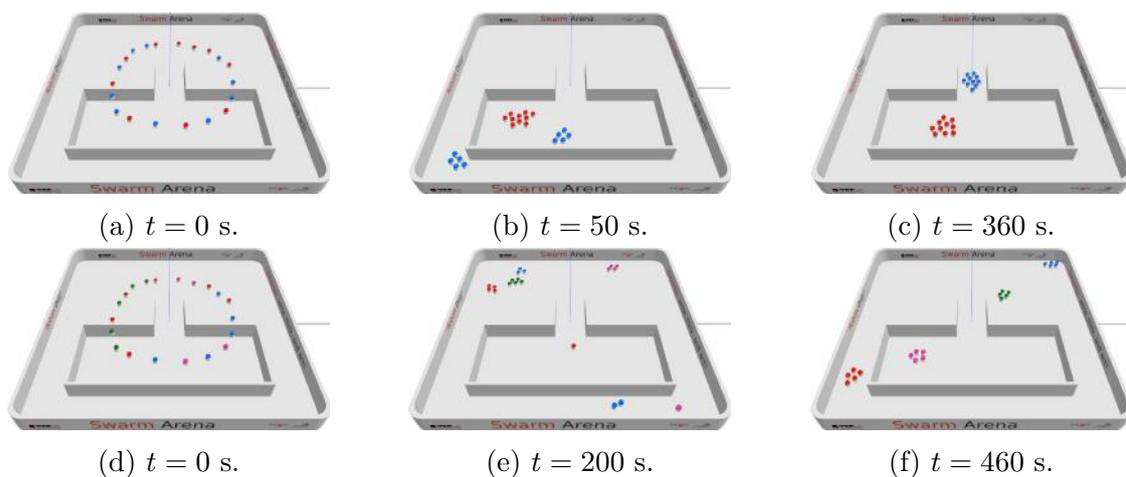


Figure 5.9: Flocking-segregative behavior demonstrated in simulated environment  $\mathcal{E}$ : (a)-(c) depict two groups of 10 robots each, while (d)-(f) showcase four groups of 5 robots demonstrating flocking segregation in a complex environment with walls forming a room with restricted access. A video is available at <https://youtu.be/tgDvhNCQp8k>.

they also displayed the capability to navigate through tight corridors, further highlighting the efficacy of our approach in complex settings.

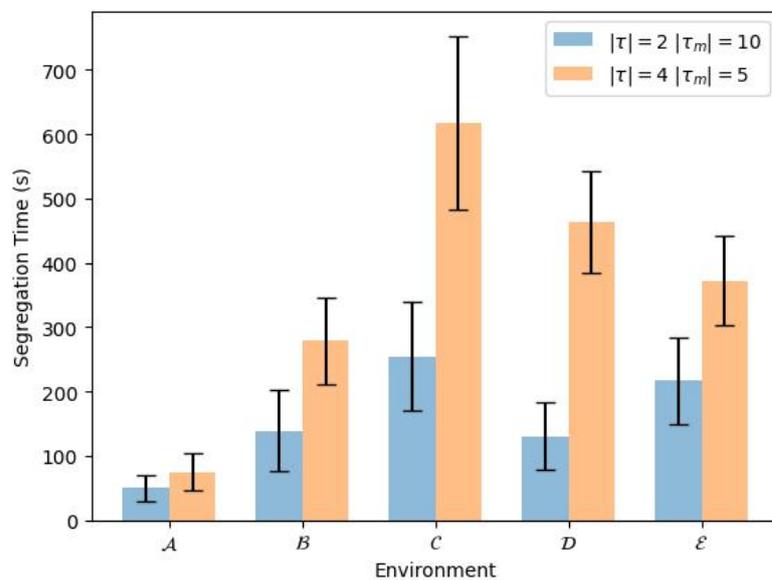


Figure 5.10: Mean convergence time to achieve segregation across diverse test environments ( $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$ ) and swarm configurations (2 groups of 10 robots, 4 groups of 5 robots) based on 15 trials.

### 5.4.4 Robustness Analysis

In this experiment, we assess the robustness of our method in face of mechanical failures occurring in one of the robots. Specifically, after a period of operation, one robot experienced motor failures, causing it to stop moving but remaining active and capable of detecting and being detected by other robots. This experiment was conducted with a group of 20 robots navigating a  $4 \times 4$  meters simulated environment in Gazebo. Upon reaching precisely 75 seconds of operation, one robot was intentionally halted to examine the potential impact on the flocking behavior. The objective was to demonstrate whether the broken robot hindered the others by disrupting the flocking dynamics. The experiment continues for a total simulation time of 220 seconds. Figure 5.11 shows a sequence of snapshots displaying the swarm’s resilience in the face of robot failure.

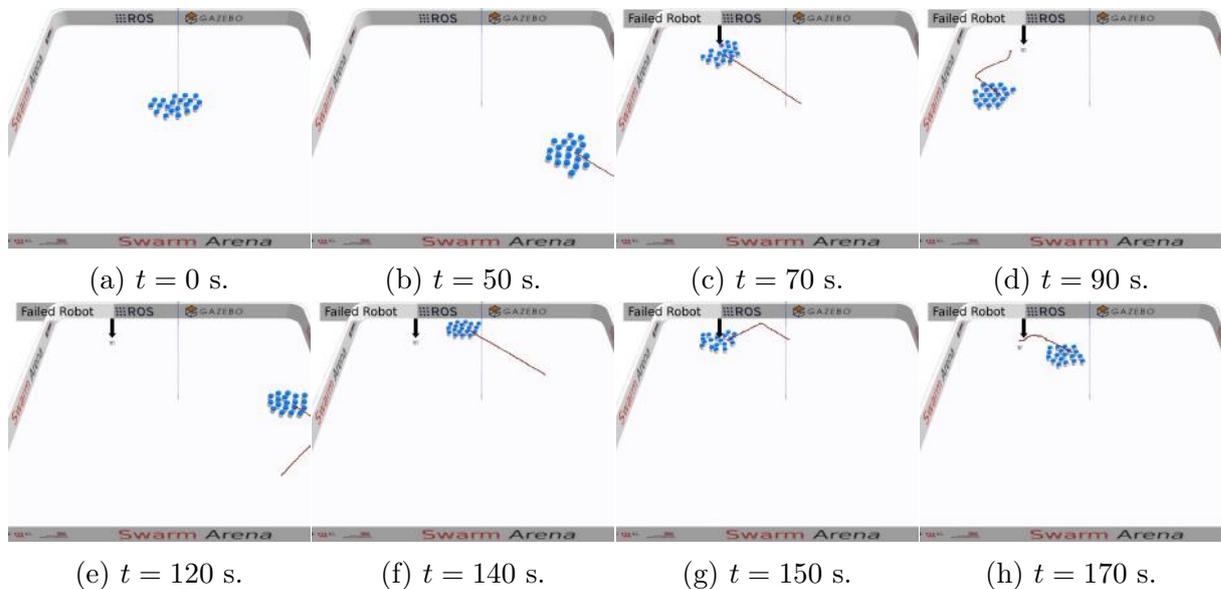


Figure 5.11: A sequence of snapshots illustrating the method’s robustness when faced with robot failure. At the 70-second mark, one robot ceases operation (indicated by the arrow). A video demonstration can be viewed at <https://youtu.be/Rq9ld4gHfo8>.

As expected, following the occurrence of the robot failure, the swarm demonstrated the ability to maintain a consensus in their group velocity, effectively leaving the broken robot behind. Although the group eventually encountered it again, they successfully reconfigured their velocities to continue flocking. Figure 5.12 depicts a plot of the average velocity error among the robots to illustrate the consensus over time.

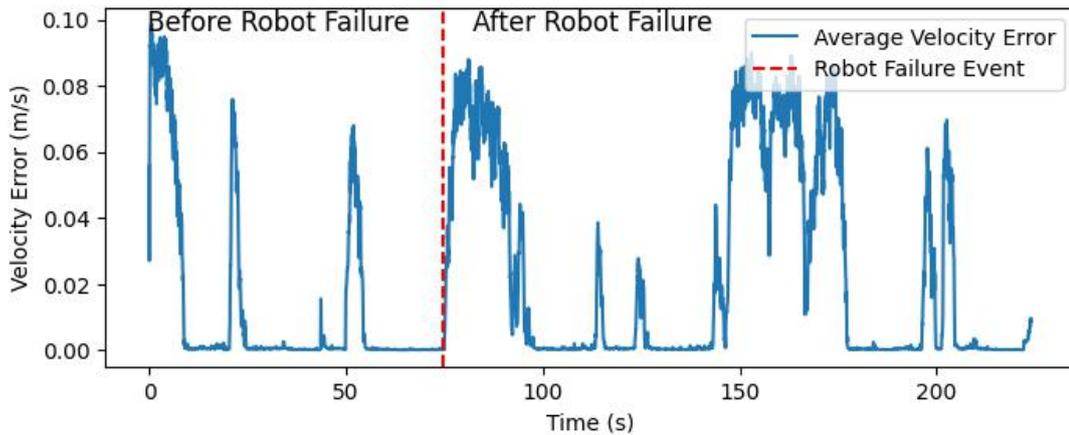


Figure 5.12: Evolution of the average velocity error among the robots to illustrate the consensus over time when faced with robot failure.

To further estimate the overall impact of robot failure on the swarm’s velocity consensus, we conducted additional trials. Specifically, we run 100 trials and register the duration the robots maintained a consensus on the average velocity below 0.005 m/s, before and after the robot’s failure event. As a result, we observe that before the robot failure, the swarm maintains consensus for  $70.2 \pm 4.6\%$  of the time, but after this, the value decreases to  $60.6 \pm 6.3\%$ . As expected, the presence of the broken robot under the specified conditions influenced the overall performance in terms of swarm consensus duration. This is attributed to the fact that the others cannot predict the broken robot behavior, introducing misinformation that affects the suitability of sampled velocity commands. Nevertheless, over time, the swarm managed to re-establish consensus in their velocities, leaving the broken robot behind. In all trials, it was observed that the swarm was never continuously hindered by the broken robot, underscoring the method’s robustness.

### 5.4.5 Real Robots

To evaluate the practical feasibility of our method in real-world settings, we conducted proof-of-concept experiments employing ten HeRo robots (described in Appendix A). These robots are commanded with velocities transmitted from a remote server executing ROS framework. To control the robots and ensure compliance with these velocity com-

mands, we employed the methodology outlined by Bruno et al. (1994), converting velocity vectors into linear and angular speeds suitable for controlling differential robots like HeRo. Given the absence of dedicated sensors for inferring the relative positions and velocities of neighboring robots, we emulate this information by extracting data from an overhead camera and employing Apriltag markers (Wang and Olson, 2016). The Apriltag markers facilitate the estimation of each robot’s pose concerning the camera. Subsequently, we leverage this pose information to compute the relative positions and velocities among the robots, thus emulating the required sensor data for our experiments.

The experimental environment is constrained within a 2 by 2 meter square, enclosed by virtual walls. We set the sensing distance to  $\lambda = 0.35$ . To explore different scenarios, we evaluate cases with one group consisting of ten robots and two groups, each comprising five robots. Figure 5.13 illustrates a single group of ten robots engaging in aggregation and flocking behaviors. Performance is quantified by metrics related to average consensus velocity and the achievement of the desired number of groups, as illustrated in Figure 5.14.

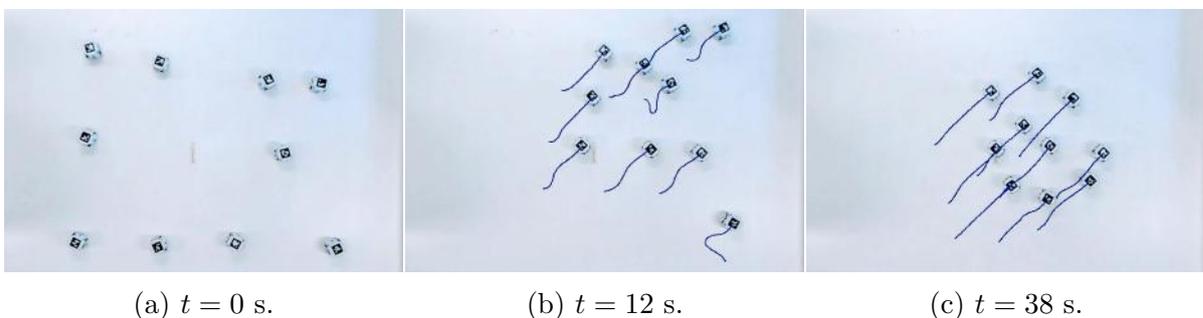


Figure 5.13: A sequence of snapshots depicting one group of ten robots aggregating and exhibiting flocking behavior. A video can be viewed at <https://youtu.be/035a6QuheMQ>.

In scenarios featuring two distinct groups, the snapshots presented in Figure 5.15 illustrate the robots engaging in segregation and demonstrating flocking behavior. The performance metrics, illustrated in Figure 5.16, show the method’s capability to attain the target group number successfully and sustain consensus in velocity. Despite the observed uncertainties inherent in real-world settings, such as noise in velocity consensus, our experiments reveal that the robots can achieve segregation and flocking behaviors.

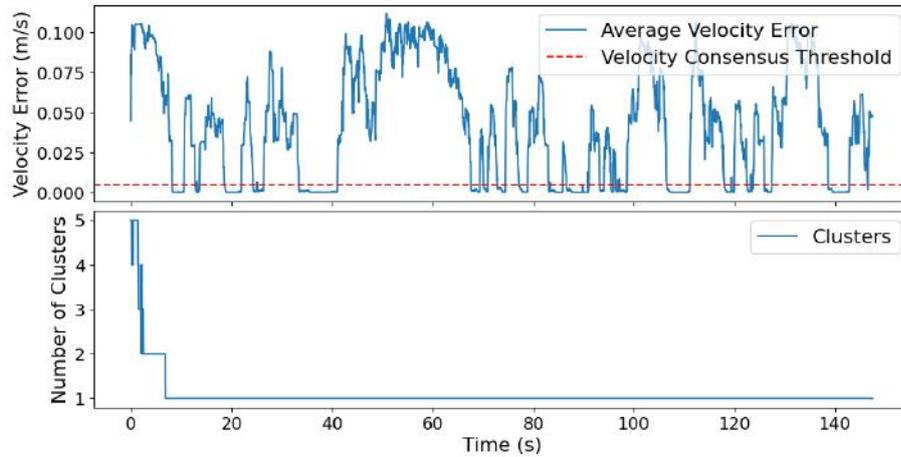


Figure 5.14: Method's performance evaluation for a single group, presenting velocity consensus and achievement of the desired number of group.

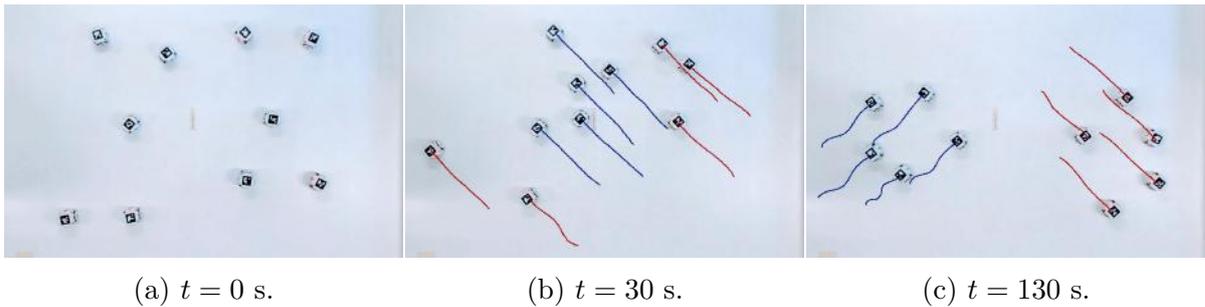


Figure 5.15: A sequence of snapshots showing two groups of five robots each, segregating and displaying flocking behavior. A video is accessible at <https://youtu.be/s1eL0mECcwc>.

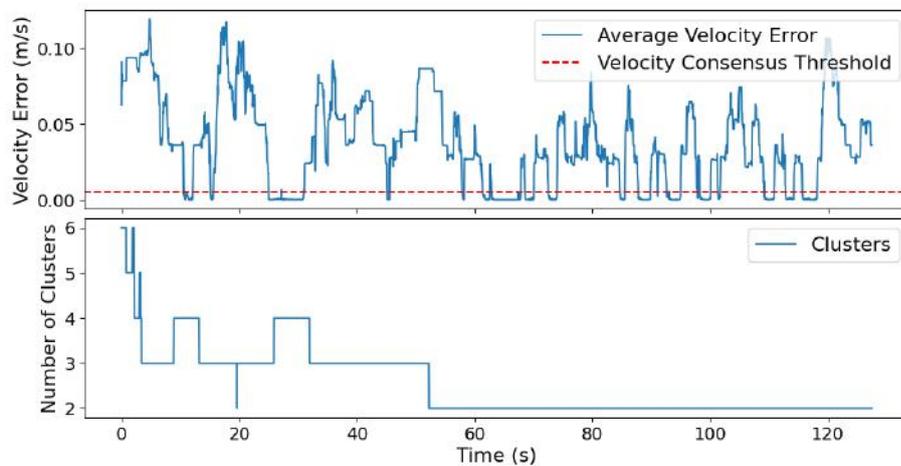


Figure 5.16: Method's performance evaluation for two groups, each comprising five robots, demonstrating velocity consensus and achievement of the desired number of groups.

# Chapter 6

## Cooperative Transport Behavior

This chapter presents another application of our methodology that allows a swarm of robots to perform a cooperative transportation task. By setting appropriate potential functions, robots can dynamically navigate, form groups, and perform cooperative transportation in a completely decentralized fashion. These behaviors emerge from the local interactions without the need for explicit communication or coordination.

### 6.1 Introduction

Robotic swarms composed of a large number of robots generally rely on emergent collective behaviors to tackle intricate challenges. These systems inherently present desirable characteristics, such as robustness, adaptability, simplicity, and scalability, which are pivotal across diverse tasks ([Tarapore et al., 2020](#); [Schranz et al., 2020](#)).

In particular, robotic swarms capable of cooperatively transporting objects may be suitable for many applications with high societal and economic impact potential. For instance, one may use robotic swarms for operations where the use of sophisticated robots is impossible or impractical, such as warehouse automation, waste disposal, and demining.

While there are numerous advantages of using swarm robots for cooperative transportation tasks, designing decentralized control methods for such applications is a challenge. Primary challenges consist of aligning and synchronizing the forces exerted to maintain effective transportation. The method must demonstrate robustness in the face



Figure 6.1: Swarm of robots cooperatively transporting an object (solid cardboard) toward a goal (transparent cardboard). The red color on the robots indicates that they are transporting the object, while the blue color indicates that the robots are looking for it.

of objects with varying shapes and resilience to changes in the environment, including surfaces with differing coefficients of friction, as well as adaptability to robot failures.

This chapter presents a method that allows a swarm of robots to navigate autonomously through a bounded environment and cooperatively transport an object toward its target location, as shown in Figure 6.1. Our approach leverages the concepts explained in Chapter 4 to model the robot swarm as a GRF and design an appropriate potential energy function as a combination of the Coulomb-Buckingham potential and kinetic energy. More especially, the Coulomb-Buckingham potential enables the robots to aggregate, interact with the object and avoid obstacles in the environment, while the kinetic energy allows the robots to reach a consensus on their relative velocities concerning their neighborhood and circulate the object looking for adequate pushing positions.

Dynamically and autonomously adapting the potential and energy parameters, robots can navigate through the environment looking for the object to be transported, form groups, and push the object toward a target location. These behaviors emerge from the local interactions without the need for explicit communication or coordination. The robots only need to be able to estimate the relative position and velocity of their neighbors and also distinguish between obstacles and the object detected within their sensing range. Moreover, the robots do not need any structural information about the object (i.e., size, mass, and shape), except for its target direction.

## 6.2 Related Work

Over the years, a wide range of control and coordination methods have been proposed to perform cooperative object transportation using multiple robots. In this section, we describe the most common transport strategies in the context of multi-robot systems, followed by a review of state-of-the-art works that especially employ the push strategy. This review underscores the distinctions and contributions of our method.

### 6.2.1 Transportation Strategies

Recently, [Tuci et al. \(2018\)](#) presented a review of the state-of-the-art works in this area, categorizing the most common transportation strategies into three types: pushing, caging, and grasping. Figure 6.2 illustrates the distinct organizational arrangements of robots associated with each strategy to sustain the transportation of the object.

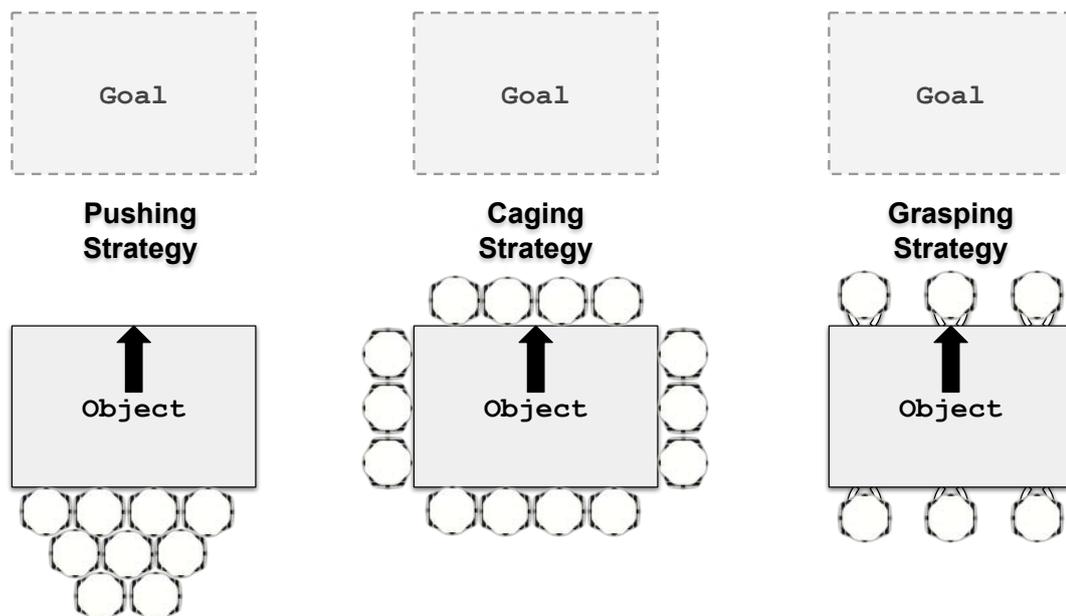


Figure 6.2: Transportation strategies using multiple robots.

Pushing strategies involve robots that are not physically attached to the object. They provide a simple way of manipulating relatively large objects and do not require

any sophisticated mechanism to apply forces to the object. Pushing is also interesting because, unlike other strategies, it allows robots to aggregate and apply forces to specific points on the object, enabling the transport of large and heavy objects when the number of robots is increased. Additionally, pushing allows for the utilization of simpler robots, making it an ideal scenario for robotic swarms.

The caging strategy can be viewed as a specialized form of pushing. It entails robots coordinating themselves to position around the object, enclosing it and sustaining this formation throughout the transport. This strategy is particularly interesting because it allows efficient maneuvers with the object, being more stable than pushing, without depending on complex mechanisms for transportation. However, it typically necessitates a specific number of robots, thereby limiting the maximum mass of the object that can be transported. Moreover, its complexity may increase depending on the object's shape, potentially requiring additional information about it.

The grasping strategy requires robots that are able to physically attach to the object and apply either pushing or pulling forces in order to transport it. This strategy allows higher dexterity during the transport of objects. However, similar to the caging strategy, it is also restricted to a specific number of robots, imposing a limit on the load capacity. Also, it may not always be suitable for swarm robots depending on the complexity of the physical mechanisms.

The strategy adopted in this dissertation consists of pushing the object towards its designated target location. We use such a strategy because it allows the use of less sophisticated robots, providing a more realistic scenario for systems with a large number of robots. Additionally, we assume the robot does not rely on direct communication systems. While, theoretically, communication among robots could enhance their collaborative capacity, in larger systems, non-communicative approaches tend to scale more seamlessly. In the subsequent section, we thoroughly review the latest state-of-the-art works based on the pushing strategy as their fundamental strategy.

## 6.2.2 Push-based Methods

While there is a large body of work in cooperative object transport (Tuci et al., 2018), we focus our discussion on approaches that do not require centralized planners or direct communication among robots to push an object from a random location to a specific goal. Table 6.1 summarizes the main characteristics of these approaches, and in the following, we review and compare these works with ours, mainly discussing scalability, adaptability, and robustness issues. By adaptability, we consider the team’s ability to work reliably even with individual robot failures or changes in the environment, while robustness is related to the power of the method to deal with objects of different mass, size, or shape.

Kube and Zhang (1993) presented one of the first studies that formally dealt with the dynamics of cooperative transport. The authors demonstrated that coordinated efforts produced by a homogeneous group of simple robots pushing an object are possible without using a direct communication mechanism. To demonstrate this hypothesis, they presented two strategies: the first one was a subsumption-style behavior-based controller with a fixed priority behavior arbitration; and the second one using reflexive behaviors and adaptive logic networks, trained with a supervised training algorithm, for behavior arbitration. Further, Kube (1997) extended the original work by constructing a new robot for experimentation. Later, Kube and Bonabeau (2000) proposed the addition of a stagnation recovery strategy avoiding deadlock conditions in which the robots cancel the pushing forces on the object depending on where they are positioned around it. Moreover, they extended their approach, allowing the robots to push the object towards a fixed goal position and performed experiments to test their method’s robustness and scalability.

Yamada and Saito (2001) also presented theoretical and experimental analyses to support the assumption that robots can transport objects without using direct communication. Unlike previous works, the authors demonstrated that the use of indirect communication also allows the transport of an object towards a target location considering a dynamic environment. Their approach consists of using an action selection method



to design a behavior-based control method that is robust to a small change in the environment, such as increase the number of robots or the mass of the object. To evaluate the performance of their approach, the authors executed real experiments using four robots showing that the robots can operate in a simple environment where individual robots are required to push a light object or in complex environments where multiple robots are required to push a heavy object cooperatively. Although their approach allows adaptability to small changes in the environment, the authors did not evaluate the robustness of the method to objects with different shapes, masses, and sizes. Furthermore, the method does not seem to scale easily as the system grows.

[Fujisawa et al. \(2013\)](#) presented a cooperative transport approach that uses an interesting mechanism for indirect communication via artificial pheromones as seen in ants. Such mechanism allows the robots to sense and lay on the terrain a volatile alcohol substance, mimicking the effect of pheromone during trail formation. The authors also proposed a behavior-based algorithm using a deterministic finite automaton, which allows the robots to perform a random search to find a food item (i.e., a heavy object), and to transport it to a target location (i.e., the nest). To evaluate the efficiency of the proposed system, the authors perform experiments with up to 10 robots demonstrating the adaptability and robustness of their approach. They showed that the system remains stable when the object's weight is changed and when considered failures in some robots, incapacitating them from carrying out the task. Concerning experiments that assess the impact of using the pheromone-based communication mechanism, they show that such a mechanism is effective only with a relatively small number of robots in the environment. For a larger group of robots, they observed an irrelevant impact on the completion time, as many robots are likely to find the food and begin the cooperative transport before the trail is formed. Although the authors present experiments that evaluated the adaptability and robustness of the proposed method, the work does not assess the system's scalability and also the robustness of the method as to the shape and size of the object changes.

[Chen et al. \(2015\)](#) proposed another behavior-based strategy for cooperative transport that deals with goal occlusion by keeping the robots pushing the object even when

not detecting the goal. They conducted experiments using twenty robots to transport objects of different shapes and provided analytical proof of the method’s effectiveness. Also, the authors demonstrated other interesting experiments in which they consider the target location as a mobile robot remotely controlled by a human, showing potential applications in the context of human-robot interaction. The work was further extended with this focus by [Kapellmann-Zafra et al. \(2016\)](#). Although the authors demonstrated the system’s scalability and robustness to different types of objects, tests evaluating the system’s adaptability to failures or environmental changes were not performed.

Differently from previous approaches, [Alkilabi et al. \(2017\)](#) proposed the use of an automatic control method to demonstrate that effective coordination of forces to transport heavy objects to an arbitrary direction can be obtained by a group of robots equipped with a minimalist sensory apparatus, and with no means of direct communication, allowing the robots to only perceive the movements of the object. The robot controller is composed by a continuous time recurrent neural network and evolutionary algorithm is employed to set its parameters. The best instance evolved of the controller was extensively tested on physical robots to transport objects of different sizes and masses to an arbitrary direction. Further, [Alkilabi et al. \(2018\)](#) presented a complementary study extending their neural controller with mechanisms to direct the transport towards a specific target location. The authors show that the transport strategies are scalable with respect to the group size, and robust enough to deal with boxes of various masses and sizes. Despite this work presenting an interesting solution and performed a satisfactory set of experiments, the strategy still requires that robots start with a direct view of the object, and also does not consider objects with different shapes or obstacles in the environment.

In the following, we present our method that allows a swarm of robots to coordinate their motion to navigate a bounded environment and cooperatively transport an object toward its target location. Different from other methods, we do not require pre-synthesized behaviors or automatic learning methods. Using appropriate potential functions, the desired behaviors emerge directly from local interactions, bringing scalability, robustness, and adaptability to our method.

## 6.3 Method

The method outlined in this chapter builds upon the concepts explained in Chapter 4, utilizing them to formulate a suitable potential energy function. This function implicitly conduces a swarm of robots to cohesively navigate through a bounded environment and cooperatively transport an object toward its target location. In the following, we introduce formal definitions and delineate our contributions to the transportation task.

### 6.3.1 Formalization

In this method, we assume a set  $\mathcal{R}$  comprising  $\eta$  homogeneous robots navigating through a bounded region within the two-dimensional Euclidean space. Equipped with a circular sensing range restricted by a radius of  $\lambda$ , these robots can measure the relative position and velocity of other robots, as well as obstacles (as outlined in Definition 4.1). In this context, we extend the sensor capabilities to include the detection of objects, and the robot interprets objects in a manner analogous to obstacles<sup>1</sup>. Consequently, we require the  $i$ -th robot to be able to arrange the points detected on objects and obstacles at time  $t$  into two distinct sets,  $\mathcal{O}_i^{(t)}$  and  $\mathcal{W}_i^{(t)}$ , respectively.

**Definition 6.1** (Objects). *An object within the environment is represented as a finite set of points  $\mathcal{O} = \{\mathbf{o}_1, \dots, \mathbf{o}_n\}$  outlining the object perimeter. An object detected by the  $i$ -th robot within its circular sensing range at time step  $t$  consists of a subset of points  $\mathcal{O}_i^{(t)} \subset \mathcal{O}$  in the robot frame, where*

$$\mathbf{o}_j \in \mathcal{O}_i^{(t)} \rightarrow \|\mathbf{o}_j - \mathbf{p}_i\| \leq \lambda,$$

and  $\|\mathbf{o}_j - \mathbf{p}_i\|$  is the Euclidean distance between the  $i$ -th robot and  $j$ -th point, assuming no other robots can block the detection of the object points.

---

<sup>1</sup>Note that the robots can differentiate among obstacles, objects and other robots using their sensors.

**Assumption 6.1** (Object Target Direction). *We assume that the robots can track the target direction  $\mathbf{g}$  of object  $\mathcal{O}$  concerning their reference frame anywhere within the environment. This implies that the robot does not require the exact target location of the object but only a normal vector  $\|\mathbf{g}\| = 1$ , denoting its direction. Once the object reaches its target location, the robot should stop sensing the target direction, such that  $\|\mathbf{g}\| \triangleq 0$ . There exist several technologies that may be used to achieve such sensing information. For instance, robots may be equipped with an array of microphones or photoresistors capable of discerning the direction of a sound or light source positioned at the target location. When the object reaches its target location, it can occlude or inhibit the propagation of the source signal to the robots. Naturally, the methodology outlined in Chapter 4 can be adjusted to accommodate other forms of sensing that grant the robot the ability to localize both the object and its target location. This would result in improved task performance but at the expense of increased computational complexity. For the purpose of this assumption, we concentrate solely on knowledge of the object's target direction, aiming to investigate a minimalist approach conducive to cooperative transport.*

**Remark.** *It is important to highlight that our method adheres to a minimalist philosophy, meaning that we operate under the assumption of a complete absence of any prior information about the object. This encompasses aspects such as its specific location, shape, mass, center of mass, and any other related attributes, with the sole exception being knowledge of its target direction relative to each robot.*

### 6.3.2 Swarm Behaviors and Potential Functions

As demonstrated in the preceding method detailed in Chapter 5, one of the notable advantages of our methodology is its inherent flexibility to accommodate various swarm behaviors. This adaptability involves properly designing the potential functions used by

the potential energy (5.1),

$$H_i(\mathbf{v}_i^{(t+1)}, \mathcal{N}_i^{(t)}) = U_{\{i\}}(\mathbf{v}_i^{(t+1)}) + \sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)}).$$

In this method, we extend the definition of the potential function  $U_{\{i\}}(\mathbf{v}_i^{(t+1)})$  to encompass interactions between robots and objects while addressing collision avoidance with obstacles within the environment. Additionally, the potential function  $U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)})$  is employed to sustain cohesive navigation among the robots within the environment.

More specifically, the Coulomb-Buckingham potential is used as an interaction factor among robots as well as their interaction with objects and obstacles. The Kinetic energy allows the robots to maintain a consensus on their velocities while navigating as a group and moving around the object. Figure 6.3 presents a diagram illustrating how the desired behaviors emerge from the combination of Coulomb-Buckingham potential and kinetic energy. These behaviors are further delineated in the subsequent sections.

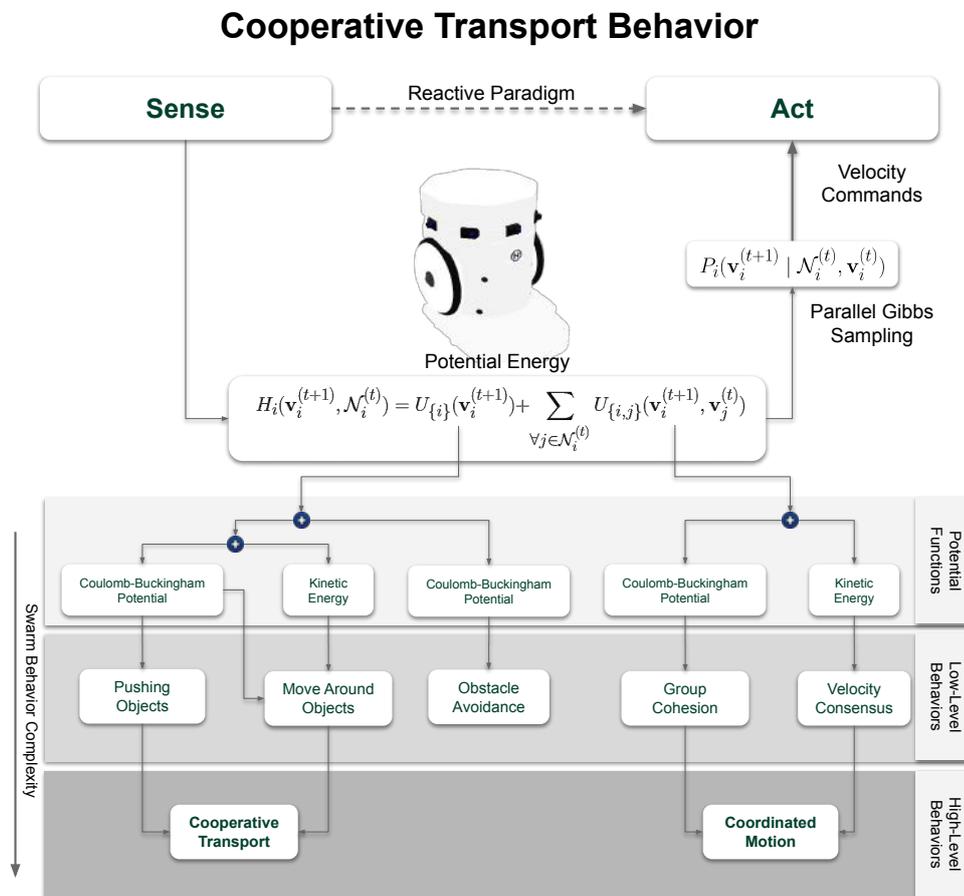


Figure 6.3: Diagram showing how the swarm behaviors emerge from the combination of the potential functions.

### 6.3.3 Cooperative Transport

The cooperative transportation task presents a significant challenge, particularly within a robot swarm, as it requires the integration of various behaviors to locate and push the object to its goal. Specifically, we require the robots to adeptly form groups, navigate cohesively, and, upon encountering an object, push it in an appropriate direction. In cases where the robot’s pushing position is unsuitable for a direct push, the robots must instead navigate around it to achieve an appropriate pushing position. In this section, we delve into the transportation behavior when the  $i$ -th robot detects the object, that is, when  $|\mathcal{O}_i^{(t)}| > 0$ . To address this behavior, we engineer the potential function  $U_{\{i\}}(\mathbf{v}_i^{(t+1)})$  to concurrently induce both move-around and pushing behaviors.

The pushing behavior directly stems from the Coulomb-Buckingham potential, which inherently induces attractive interactions. Conversely, the move-around behavior necessitates the robot to outline the object’s contours while maintaining a safe distance to prevent collisions. Achieving this requires a combination of the Coulomb-Buckingham potential with kinetic energy, as the first induces the robot towards a suitable distance while the second allows the object’s contour to dictate the robot’s motion. It is important to highlight that the move-around and pushing behaviors do not emerge simultaneously. Upon reaching an appropriate pushing position, the pushing component should be more prominent over all other influences. Before going into the specifics of this conditional factor, it is necessary to understand the underlying move-around and pushing behaviors.

Formally, the move-around behavior is induced by combining the Coulomb-Buckingham potential with kinetic energy. The first potential forces the robots to maintain a specific distance from the object, defined as  $\frac{\lambda}{2}$ , preventing missing or colliding with it. To achieve this, we set the minimum energy distance  $r_0 = \frac{\lambda}{2}$  and adopt a heterogeneity factor  $C(i, j)$  that is strictly negative in Equation (4.7). The second potential determines which velocities are conducive to circumnavigating the object. The strategy entails the computation of a gradient derived from the points detected on the object’s surface.

To compute the gradient, we arrange the points detected by the  $i$ -th robot on the

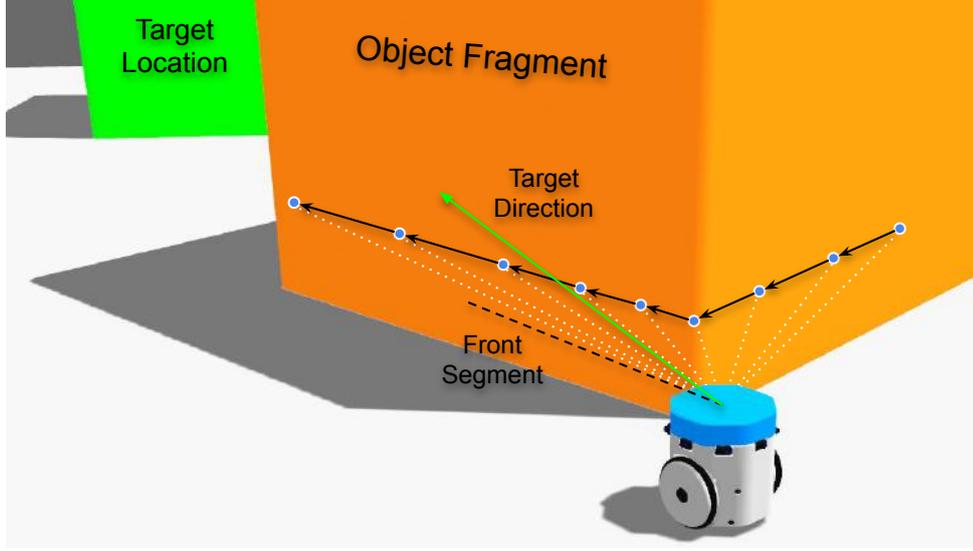


Figure 6.4: Robot's perception regarding the object and its target location.

$\mathcal{O}_i^{(t)}$  object on a clockwise or counterclockwise order. To decide in which order, we project a line segment (“front segment” depicted in Figure 6.4) and count the number of points to the left and to the right of that line segment. If there are more points to the left than right, we order them clockwise; otherwise, we order them counterclockwise. Practically, this forces the robots to choose velocities that align them with the object's surface. For now on, assume  $\bar{\mathcal{O}}_i^{(t)}$  is the ordered set of points detected on the object.

Defining the order of the elements directly implies the direction of the gradient, which, in turn, indicates the direction the robot should move around the object. Given  $\bar{\mathcal{O}}_i^{(t)}$ , the gradient is calculated as follows:

$$\nabla \bar{\mathcal{O}}_i^{(t)} = \{\mathbf{o}_{i+1} - \mathbf{o}_i : |\bar{\mathcal{O}}_i^{(t)}| \geq 2\}, \quad (6.1)$$

where  $\nabla \bar{\mathcal{O}}_i^{(t)}$  is a set containing vectors located on the object's surface, each detected within the sensing range  $\lambda$  and referenced to the robot's frame.

To evaluate how much the velocity performed by the robot favors the move-around behavior, we sum the differences between the gradient vectors and the robot's instantaneous velocity. That is, we compute the resulting vector,

$$\mathbf{Q}_i = \sum_{\forall j \in \nabla \bar{\mathcal{O}}_i^{(t)}} \nabla \mathbf{o}_j - \mathbf{v}_i^{(t+1)}, \quad (6.2)$$

where  $\mathbf{v}_i^{(t+1)}$  is a prospect instantaneous velocity for the  $i$ -th robot, while  $\mathbf{Q}_i$  is a vector that encapsulates the disparities between this velocity and the gradient.

Finally, given the strategies and definitions outlined above, we formally express the potential function  $U_{\{i\}}(\mathbf{v}_i^{(t+1)})$  that induces move-around behavior as follows:

$$U_{\{i\}}(\mathbf{v}_i^{(t+1)}, \mathcal{O}_i^{(t)}) = \sum_{\forall j \in \mathcal{O}_i^{(t)}} \Phi(\|\mathbf{p}_i^{(t+1)} - \mathbf{o}_j\|, r_0) + \mathbf{E}_k(\mathbf{Q}_i), \quad (6.3)$$

where  $\mathbf{p}_i^{(t+1)}$  is the projected position calculated using the motion model  $\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)})$ ;  $r_0$  is the minimum energy distance, which is set to  $r_0 = \frac{\lambda}{2}$ ; and  $\forall j \in \mathcal{O}_i^{(t)} : C(i, j) < 0$ .

Now that we have established the move-around behavior, which involves the robots circumnavigating the object, the next step is to incorporate a pushing behavior into the potential energy defined in Equation (6.3). This can be achieved by reducing the minimum energy distance  $r_0$  in the Coulomb-Buckingham potential. If  $r_0$  is set sufficiently short, this adjustment will compel the robots to approach and collide with the object, resulting in a pushing behavior. However, the challenge lies in arranging a conditional factor that properly adjusts  $r_0$  to determine whether the robot pushes or moves around the object.

To engineer this conditional factor, we define  $r_0$  as a function,  $r_0(\cdot)$ , that conditions the robot's distance relative to the object. This involves partitioning the object points,  $\mathcal{O}_i^{(t)}$ , into left,  $\mathcal{O}_{i,l}^{(t)}$ , and right,  $\mathcal{O}_{i,r}^{(t)}$ , points relative to the target direction vector, depicted by the green arrow in Figure 6.4. When points exist on both sides, i.e.,  $\|\mathcal{O}_{i,l}^{(t)}\| > 0$  and  $\|\mathcal{O}_{i,r}^{(t)}\| > 0$ , it indicates that the object is occluding the goal from the robot; otherwise, the robot has a direct line of sight to the goal. The decision to start pushing the object depends on whether points are present on both sides, and the proportion of points from one side to the other is below a threshold  $\rho$ . Formally, the conditional factor is defined as

$$r_0(\mathcal{O}_i^{(t)}) = \mathbb{1} \left( \left| \min \left( \frac{\|\mathcal{O}_{i,l}^{(t)}\|}{\|\mathcal{O}_{i,r}^{(t)}\|}, \frac{\|\mathcal{O}_{i,r}^{(t)}\|}{\|\mathcal{O}_{i,l}^{(t)}\|} \right) - \frac{1}{2} \right| > \rho \right) \frac{\lambda}{2} + \epsilon, \quad (6.4)$$

where  $\mathbb{1}(\cdot)$  is the indicator function,  $\lambda$  represents the sensing range and  $\epsilon \ll \frac{\lambda}{2}$ . The idea is that when the robot is suitably positioned to push the object toward the goal, this function converges towards zero, significantly elevating the Coulomb-Buckingham potential. Conversely, in cases where a safe distance is required to prevent collision, the function sets a distance that ensures the robot remains suitably distanced from the object.

Finally, inducing both move-around and pushing behaviors only requires substituting  $r_0$  in Equation (6.3) with the conditional factor implemented by the function  $r_0(\mathcal{O}_i^{(t)})$ .

In the subsequent sections, we elucidate how additional behaviors are incorporated into the potential energy formulation.

### 6.3.4 Obstacle Avoidance

To ensure collision avoidance with obstacles, we augment the potential  $U_{\{i\}}(\mathbf{v}_i^{(t+1)})$  by incorporating an additional Coulomb-Buckingham potential. This extension involves computing the potential based on the distance  $r$  between the points detected on the surface of obstacles,  $\mathcal{W}_i^{(t)}$ , and the robot's position. By assigning positive charges,  $C(i, j) > 0$ , we induce the robots to exhibit a repulsive behavior towards the obstacles, effectively preventing collisions. Formally, we extend the potential function as follows:

$$U_{\{i\}}(\mathbf{v}_i^{(t+1)}, \mathcal{O}_i^{(t)}, \mathcal{W}_i^{(t)}) = \sum_{\forall j \in \mathcal{O}_i^{(t)}} \Phi(\|\mathbf{p}_i^{(t+1)} - \mathbf{o}_j\|, r_0(\mathcal{O}_i^{(t)})) + \mathbf{E}_k(\mathbf{Q}_i) + \sum_{\forall j \in \mathcal{W}_i^{(t)}} \Phi(\|\mathbf{p}_i^{(t+1)} - \mathbf{w}_j\|). \quad (6.5)$$

In this expression, the first term represents the potential derived from object interaction, with  $r_0$  dynamically determined based on the object's relative position. The second term accounts for the kinetic energy aspect, ensuring robots circumnavigate the object's contour. The last term introduces the repulsive potential associated with obstacles, influencing the robots to maintain a safe distance. The summation is performed over all points on both the object and obstacle surfaces. This extended potential formulation addresses both cooperative transport and collision avoidance behaviors.

### 6.3.5 Cohesive Motion

To induce cohesive motion within the swarm, we establish the potential function  $U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j^{(t)})$  as a combination of the Coulomb-Buckingham potential and kinetic energy. This approach compels the robots to aggregate while simultaneously working

towards a consensus on the group's velocity. The formulation is identical to that presented in Chapter 5 and is expressed formally by the potential function in Equation (5.3),

$$\sum_{\forall j \in \mathcal{N}_i^{(t)}} U_{\{i,j\}}(\mathbf{v}_i^{(t+1)}, \mathbf{v}_j) = \sum_{\forall j \in \mathcal{N}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathcal{K}(\mathbf{p}_j^{(t)}, \mathbf{v}_j^{(t)})\|) + \mathbf{E}_k(V_i^{(t)}).$$

In this formulation, the Coulomb-Buckingham potential induces the robots within the swarm to form groups while the kinetic energy component guides them toward a shared velocity. This integration of potentials instigates a cohesive motion, promoting synchronized movement among the robots.

### 6.3.6 Combination

Finally, we consolidate the potential functions developed earlier (as depicted in Figure 6.3) into the overall potential energy formulation, expressed by Equation (5.1). This results in the following potential energy expression,

$$H_i(\mathbf{v}_i^{(t+1)}, \mathcal{N}_i^{(t)}, \mathcal{W}_i^{(t)}, \mathcal{O}_i^{(t)}) = \left( \sum_{\forall j \in \mathcal{O}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{o}_j\|, r_0(\mathcal{O}_i^{(t)})) + \mathbf{E}_k(\mathbf{Q}_i) \right) + \left( \sum_{\forall j \in \mathcal{W}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{w}_j\|) \right) + \left( \sum_{\forall j \in \mathcal{N}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathcal{K}(\mathbf{p}_j^{(t)}, \mathbf{v}_j^{(t)})\|) + \mathbf{E}_k(\mathbf{V}_i) \right), \quad (6.6)$$

where the first term within the parentheses encodes the move-around and pushing behaviors essential for inducing the cooperative transport behavior. The second term produces obstacle avoidance behaviors, while the third promotes coordinated motion behaviors.

After establishing the potential energy function, we proceed to compute the conditional probability function in Equation (4.6) using the Algorithm (1), as detailed in Chapter 4. This computation yields the most likely instantaneous velocity for each robot, encoded by these different swarm behaviors. The prominence of a specific behavior depends on the robot's sensing state. Through continuous iterations, the swarm dynamically transitions between different behaviors, effectively promoting the transportation task.

## 6.4 Experiments and Results

In order to complete evaluating the method, we conducted a series of simulated experiments to assess the performance of our method in terms of scalability, adaptability, and robustness. For this, we implemented our method using the ROS<sup>2</sup> middleware and set up a simulated environment in Gazebo<sup>3</sup>. The simulated environment consists of a 4 by 4 meters arena, an object, and a group of differential drive robots. These simulated robots are a model of a physical robot that we developed in our laboratory for experiments with swarm robotics (described in Appendix A). It is equipped with a range sensor consisting of a set of infrared beams with  $\lambda = 0.5$  meters, mounted on top of the robot to prevent occlusion of the object view by other robots. We consider that this sensor can distinguish between objects and obstacles. As mentioned, we also consider that robots can estimate neighbors' relative positions and velocities.

To control the robots and ensure compliance with desired velocity commands, we employed the methodology outlined by Bruno et al. (1994). This approach is particularly adept at converting velocity vectors into linear and angular speeds, suitable for controlling differential robots like HeRo. The maximum linear speed reached by these robots is 0.12 m/s. The simulator's physics engine allows a single robot to start the movement of a 200 grams object if it collides with it at speeds above 0.10 m/s. It also allows it to keep pushing the object if it maintains a speed greater than 0.01 m/s. For each experiment, we performed 30 runs and, for each run, the robots are randomly placed in the environment. The results are presented as an average value with a 95% confidence interval. A video of the experiments is available on Youtube<sup>4</sup> and the source code at Github<sup>5</sup>.

---

<sup>2</sup>Robot Operating System. ROS. <https://www.ros.org>.

<sup>3</sup>Gazebo. <http://gazebo.org>.

<sup>4</sup>Experiments videos: <https://youtu.be/hrkJKL3W3pQ>.

<sup>5</sup>Source code: [https://github.com/rezeck/grf\\_transport](https://github.com/rezeck/grf_transport).

### 6.4.1 Scalability Analysis

To assess the system’s scalability, we conducted experiments increasing the number of robots: 2, 4, 10, 20. In each scenario, a rectangular object measuring 0.5 by 0.4 meters and weighing 200 grams was positioned at the center of the arena, with the target location situated 1.7 meters away. It is assumed that the robots stop pushing the object if it approaches within 0.1 meters of its target location, indicating the robot stops perceiving the target direction. Figure 6.5 showcases sequential snapshots from a representative experiment for each robot count. Subsequently, Figure 6.6 presents the results, and Table 6.2 summarizes the average time the robot takes to transport the object to its goal.

Table 6.2: Average transportation time for 30 trials when the number of robots increases.

Number of robots	Transport time (seconds)
2	$829 \pm 74$
4	$593 \pm 96$
10	$324 \pm 18$
20	$278 \pm 14$

As expected, increasing the number of robots reduces the time taken to find and transport the object to the target location. With more robots searching, the time taken to find the object is smaller. Additionally, the object reaches higher velocities when pushed by a greater number of robots, consequently shortening the overall transport duration.

However, it is noteworthy that the incremental advantage is smaller when employing many robots. For instance, the performance observed in transitioning from 2 to 4 robots surpasses that of moving from 10 to 20 robots moving the object. When too many robots try to push the object, the most distant ones cannot detect it due to sensing restrictions and will not contribute to the pushing. Thus, while our approach is scalable, the performance will depend on the size of the object and the robots’ sensing capabilities, and it may be saturate for many robots.

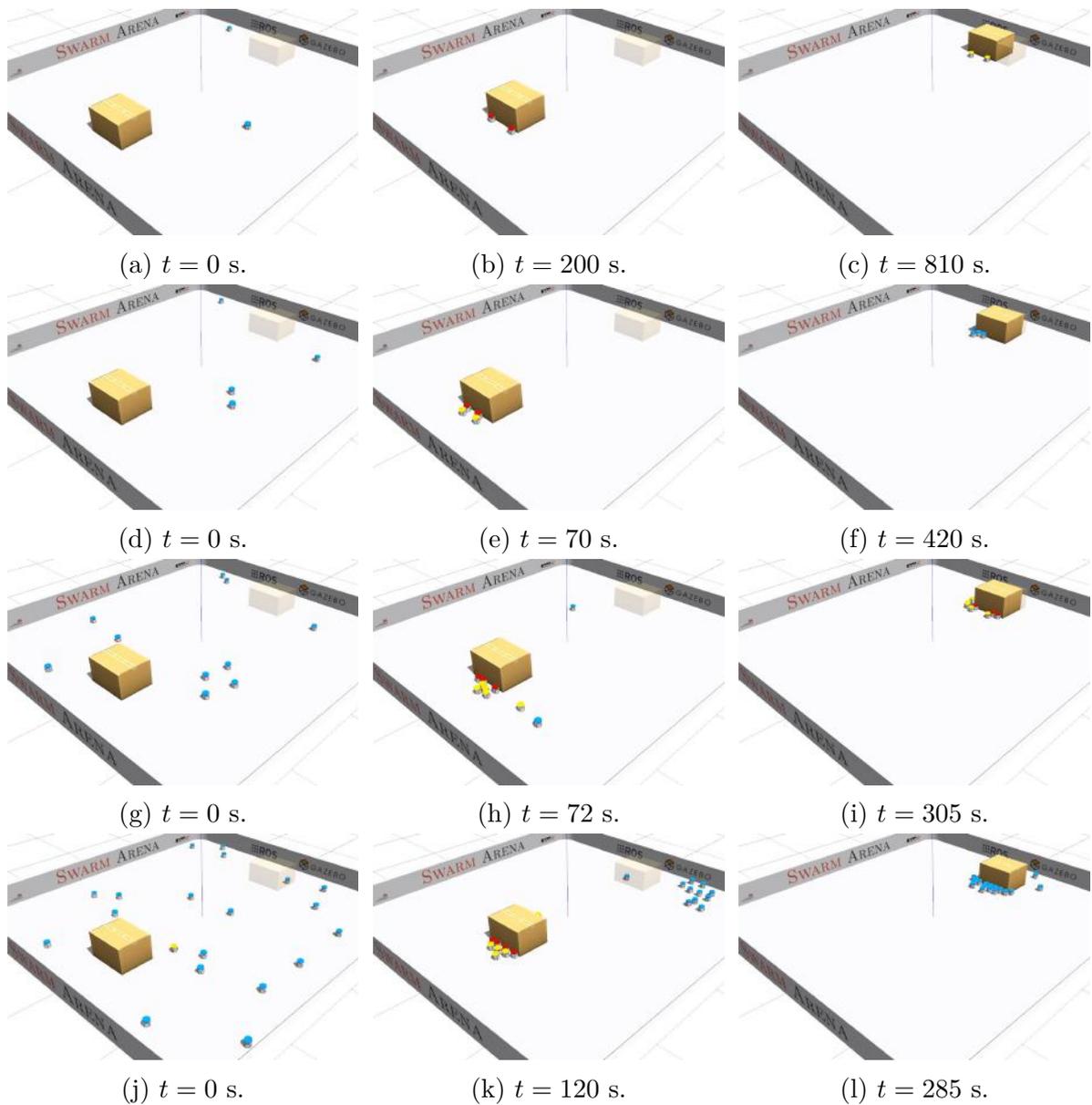


Figure 6.5: Snapshots of four distinct experiments showcasing the system's scalability with varying numbers of robots. In (a)-(c), a scenario is illustrated with a group of 2 robots, followed by (d)-(f) displaying a scenario with 4 robots. The system's performance continues to scale as depicted in (g)-(i) with 10 robots and (j)-(l) with 20 robots. Each scenario demonstrates cooperative object transportation.

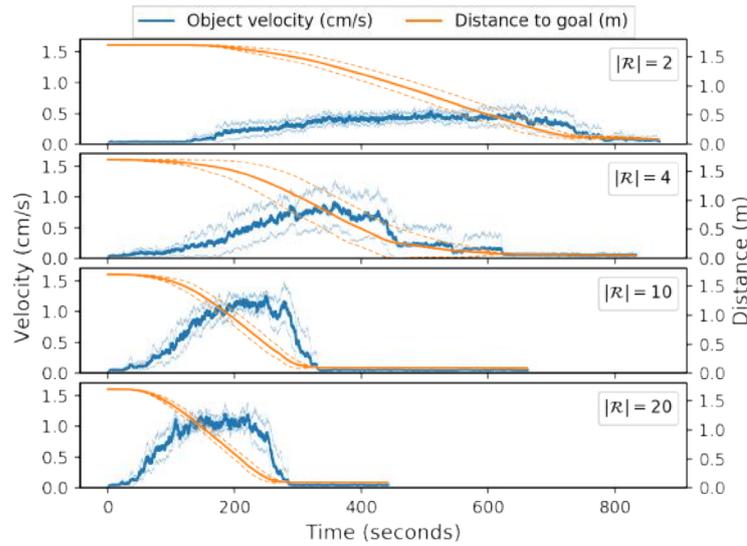


Figure 6.6: Results on the scalability of the system when increasing the number of robots.

## 6.4.2 Adaptability Analysis

To assess the adaptability of our method in the face of unexpected circumstances, we devised two experiments involving a group of 10 robots. These experiments were designed to emulate real-world scenarios where potential contingencies such as robot failures or suddenly shifts in the target location occurs. The objective was to transport a rectangular object measuring 0.5 by 0.4 meters, weighing 200 grams, to a target situated 2.6 meters away. Figure 6.7 presents a series of sequential snapshots, providing a visual account of how the swarm dynamically responded to these environmental changes.

In the initial experiment, we introduced a controlled scenario where 4 robots encountered mechanical failures while actively pushing the object. This scenario allowed us to analyze the method’s robustness in the face of decreases in the number of active robots. By subjecting the system to such a challenge, we gain insights into the overall resilience of the robots in executing the transportation task. Figure 6.8 provides results, showcasing the average velocity and the object’s distance from its target location. Complementing these results, Table 6.3 summarizes the average time spent by the robots in transporting the object. As expected, instances of robot failure had a noticeable impact on the task’s duration. Moreover, in many trials, the inactive robots obstructed their active counterparts, resulting in a notable decrease in the object’s velocity. Despite these challenges,

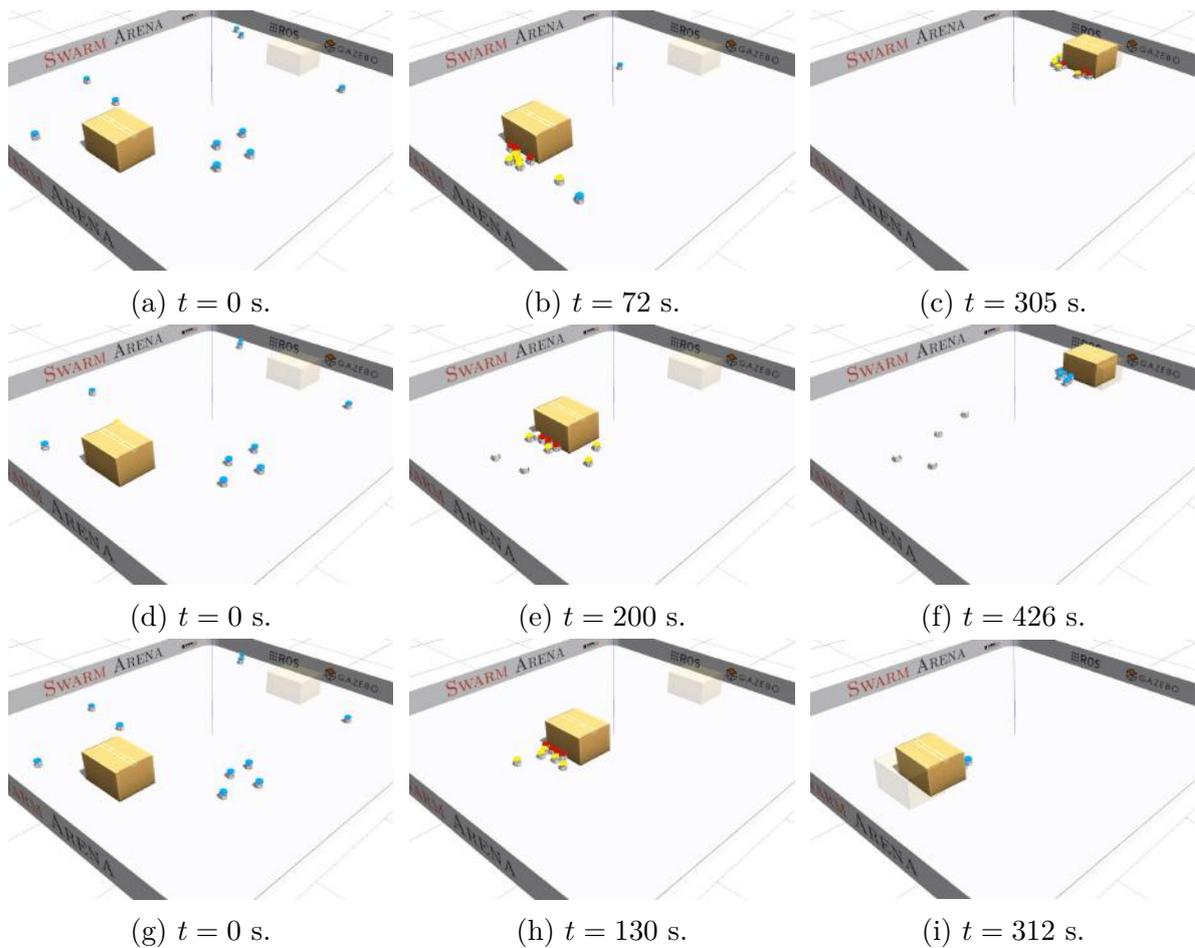


Figure 6.7: Snapshots illustrating the system’s adaptability to robot failure and dynamic target location changes.. In (a)-(c), a baseline scenario is illustrated with a group of 10 robots effectively transporting the object. Subsequently, (d)-(f) present a scenario where 4 robots stop working while pushing the object. Finally, (g)-(i) capture the moment when the target location suddenly changes, prompting the robots to change their pushing position and successfully transport the object from an alternative location.

the swarm demonstrated its capacity to successfully transport the object in all trials.

In the subsequent experiment, we evaluate the swarm’s behavior in response to dynamic alterations in the target location. To assess this experiment, we introduced a deliberate shift in the target location when the object was within a proximity of less than 1.3 meters from its initial goal. This shift entailed relocating the target to a distance of 1.3 meters in the opposite direction. This adjustment compelled the robots to circumnavigate the object to continue the transportation task. Despite this shift forcing the robots to reposition, the robots adeptly and efficiently completed the task in all trials. Notably, the time required to circumnavigate the object did not exhibit a significant increase in overall transportation time compared to the ideal scenario, thereby attesting to

the remarkable adaptability of the proposed method. Figure 6.8 and Table 6.3 display the method’s performance under this scenario.

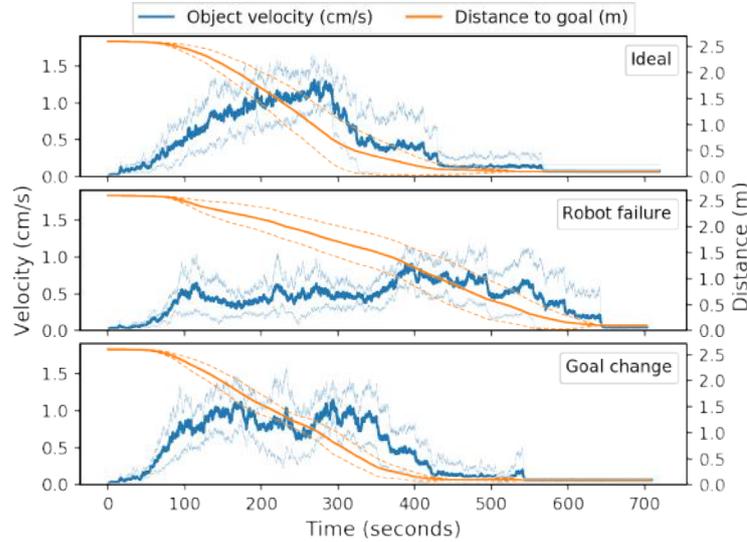


Figure 6.8: Adaptability of the swarm when robot failure occurs and when the target location changes.

Table 6.3: Average transportation time for 30 trials when the robots need to adapt to changes in the environment.

Scenario	Transport time (seconds)
Ideal	$438 \pm 13$
Robot failure	$596 \pm 21$
Goal change	$449 \pm 11$

### 6.4.3 Robustness Analysis

To thoroughly assess the method’s robustness, a series of experiments was conducted employing three distinct objects characterized by varying geometries, sizes, and masses. This diverse selection of objects included a rectangular prism with right angles, an octagonal prism featuring obtuse angles, and a triangular prism characterized by acute angles. By employing such a varied set of objects, we investigate how robots respond to pushing on varying geometric properties. Furthermore, we analyze the effects of doubling the sizes and masses of each object. Throughout all experiments, we placed the object

at a distance of 2.6 meters from its target location. Figure 6.9 displays a group of 10 robots transporting a rectangular prism, showcasing two sizes. Likewise, Figure 6.10 and Figure 6.11 illustrate the swarm’s efforts in transporting objects of octagonal and triangular configurations, respectively, each featuring two varying sizes. A synthesized overview of the experimental outcomes, encapsulating the average time required for successfully transporting these objects to their target locations, is presented in Figure 6.12. These results were summarized from a series of 30 trials per object configuration.

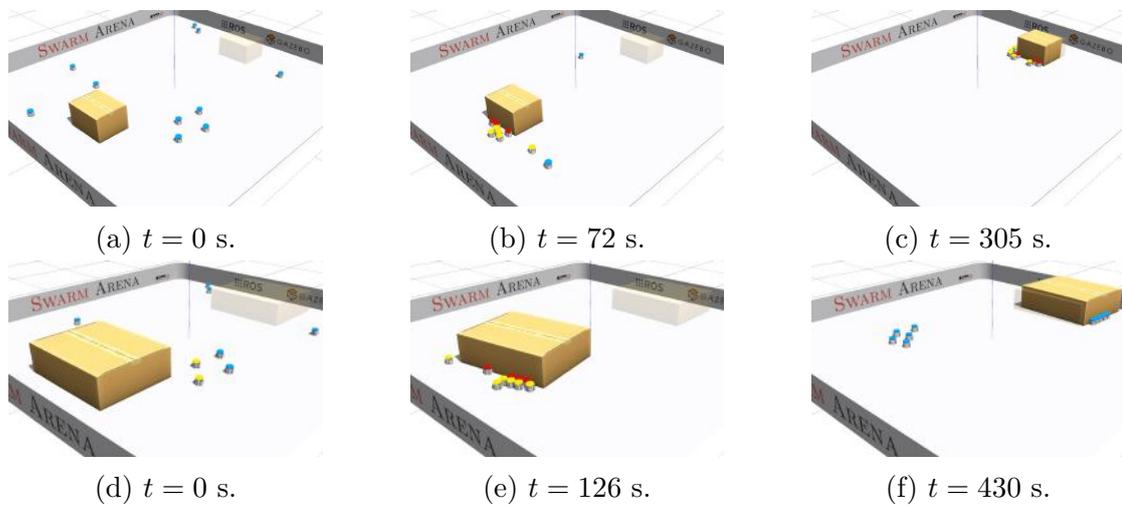


Figure 6.9: Swarm’s efforts in the transportation of rectangular prismatic objects, showcasing two distinct sizes, towards their target location.

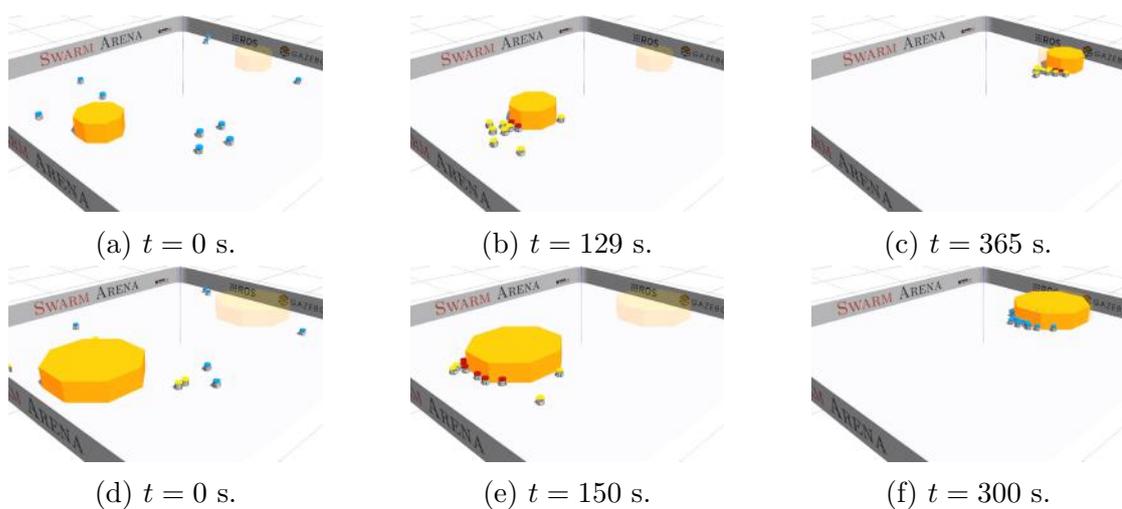


Figure 6.10: Swarm’s efforts in the transportation of octagonal prismatic objects, showcasing two distinct sizes, towards their target location.

In this experiment, the robots were less efficient in pushing the triangular prism than the other two objects. When the robots start pushing the triangle at its acute corners,

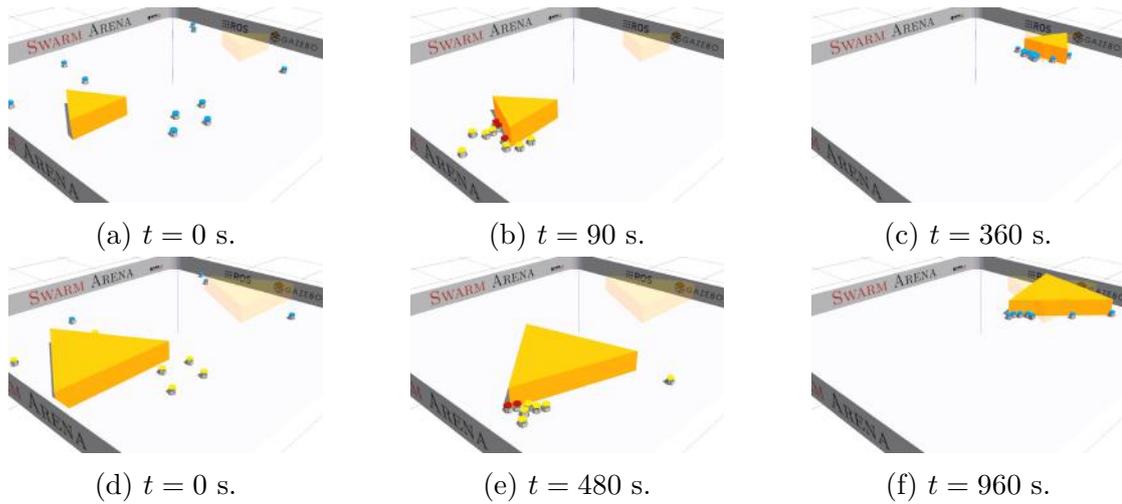


Figure 6.11: Swarm’s efforts in the transportation of triangular prismatic objects, showcasing two distinct sizes, towards their target location.

few robots are grouped to push, increasing the task time. As expected, increasing the mass of the object also increases the transport time. Although larger objects enable more robots to push it, they also cause an overall increase in transport time. Increasing the effective contact surface between the object and the ground increases the maximum intensity of the friction force, making the object more difficult for the robots to push. When we increase the size of the triangular prism, we observe that its impact over the transport time is relatively higher than increasing either the rectangular or the octagonal prism. We believe that objects with sharp corners decrease our method’s performance since we assume our sensor to be radial, making detection difficult when the robot is pushing on a corner. Despite the disparity in the transport time for different objects, the robots successfully transported all objects to their target location.

#### 6.4.4 Object Transportation in Complex Environments

In more complex environments, pushing an object toward its target location may be challenging due to obstructions caused by obstacles. Navigation strategies can help robots to move an object to its final location, passing through a series of waypoints.

Given our method’s capacity to enable robots to adapt to shifts in target locations,

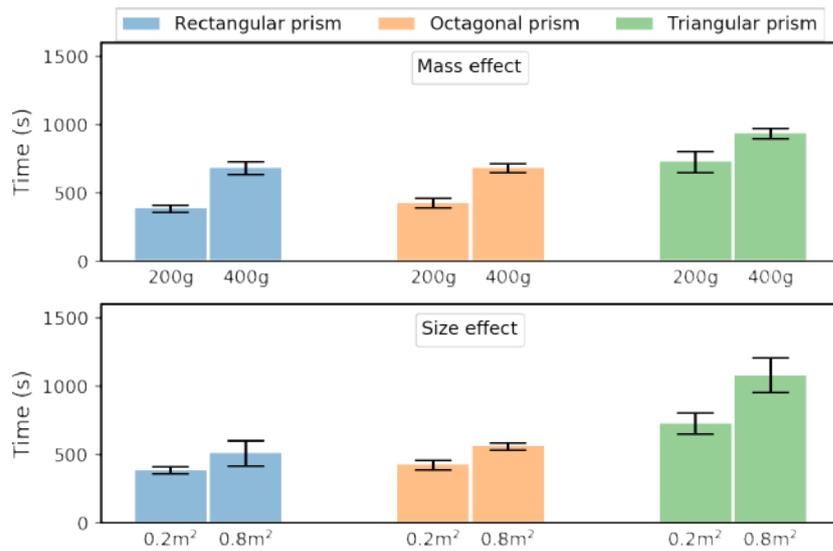


Figure 6.12: Robustness of the swarm regarding objects with different sizes and masses.

we conducted experiments involving the transportation of an object through a sequence of goals. We initiated the experiments with the assumption that the robots were initially dispersed throughout the environment, possessing no prior knowledge of the object’s location. However, they were provided with the sequence of target directions in advance. Figure 6.13 provides a sequential visual account of one such experiment, and for a more dynamic view, a video is accessible<sup>6</sup>.

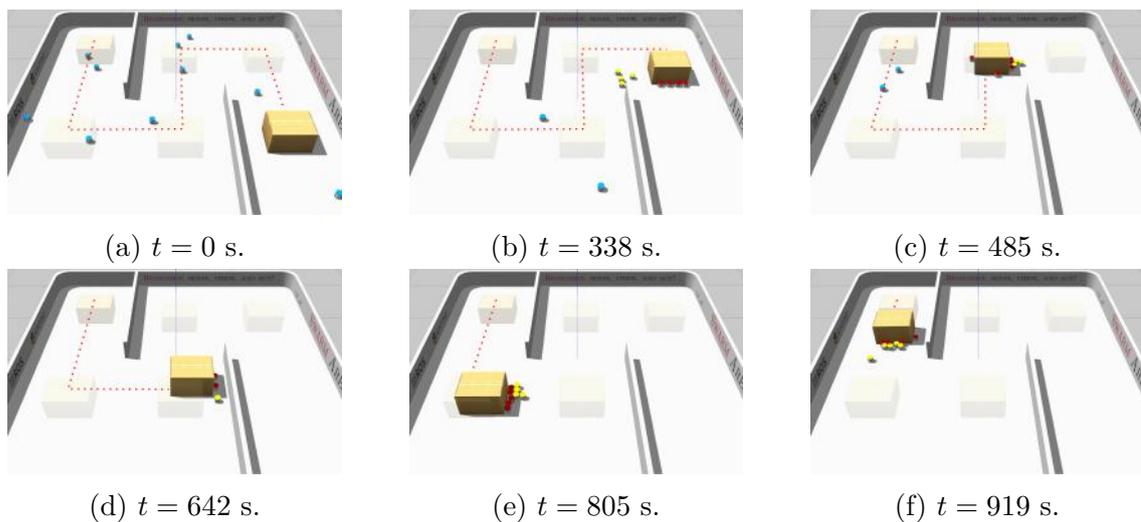


Figure 6.13: Snapshots of an experiment showing 10 simulated robots transporting an object toward a sequence of target locations in a complex environment.

<sup>6</sup>Experiment video: <https://youtu.be/a0eYZid3jQs>

### 6.4.5 Real Robots

We also conducted proof-of-concept experiments to validate the practicality of our method in a real-world setting. The experimental environment comprised a confined square area measuring 2 meters. An object, measuring 0.14 by 0.48 meters and weighing 550 grams, was placed within this space alongside five e-puck robots (Mondada et al., 2009). These robots received velocity commands from a remote server running ROS, with each functioning as an independent process. Due to the absence of sensors on our robots that would facilitate the execution of our method, we emulate these sensors using data from an Optitrack motion capture system<sup>7</sup>. The sensing range was set to  $\lambda = 0.3$ , and the target location was set at a distance of 1.1 meters in front of the object.

We executed 10 trials, each spanning a duration of 10 minutes. As we expected, the robots successfully transported the object in all instances, demonstrating the method's effectiveness even in the presence of uncertainties inherent to real-world scenarios. On average, the robots took  $448 \pm 8$  seconds to complete the transportation task. Figure 6.14 provides a sequence of visual snapshots from one such experiment.

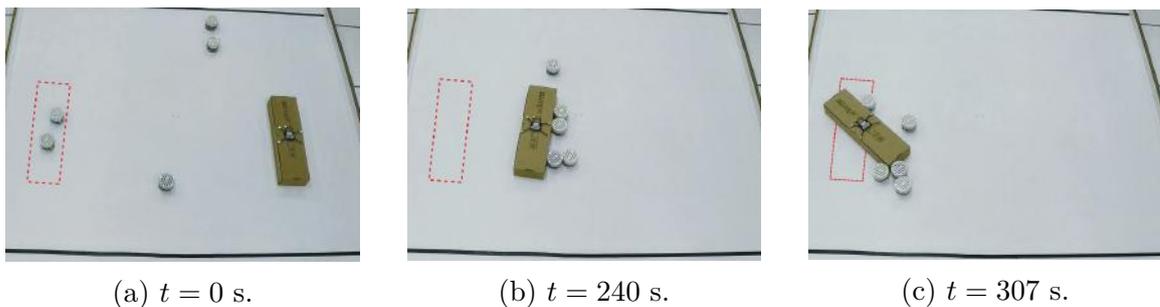


Figure 6.14: Snapshots of an experiment showing 5 e-puck robots transporting an object toward its goal. The red rectangle indicates the target location. For a more dynamic representation, a video is also accessible at <https://youtu.be/1I9-hTQ08CU>.

<sup>7</sup>NaturalPoint. Motion Capture System. <http://optitrack.com>.

# Chapter 7

## Pattern Formation Behavior

Self-organized emergent patterns can be widely seen in particle interactions producing complex structures such as chemical elements and molecules. Inspired by these interactions, this chapter presents a method that allows a swarm of heterogeneous robots to create emergent patterns in a completely decentralized fashion and relying only on local information. The method extends our methodology by incorporating additional rules on the neighborhood system. Specifically, we draw inspiration from the binding polarity rules observed in chemistry to determine which robots should interact and how they should interact with each other.

### 7.1 Introduction

Systems of a large number of particles dynamically interacting pairwise produce extraordinarily complex patterns (Saintillan and Shelley, 2008; Von Brecht et al., 2012). Well-known examples of patterns generated by these systems are molecular structures that emerge from atomic interactions depending on environmental conditions (Verwey, 1947; Gillespie and Nyholm, 1957). The study of such systems pervades many disciplines ranging from physics and chemistry to biology and pharmacy, having high societal and economic impact. In particular, one may use it to design new compounds or materials and understand biological systems.

Although challenging, the study of such systems may provide powerful tools for a

wide variety of applications in robotics, especially for swarm robotics in pattern (shape) formation problems. The pattern formation problem may be defined as the coordination of a group of robots to get into and maintain a formation with a certain shape (Bahceci et al., 2003). A key aspect for the applicability of these models in swarm robotics is the requirement for distributed and decentralized processing relying only on local information. Models with these characteristics bring several practical advantages allowing scalability, resiliency, and adaptability. Examples of potential applications would be oil spill containment or cleaning in oil plants (Kim et al., 2012; Shah et al., 2018) and constructing structures such as a temporary bridge that could dynamically adjust its size and shape to fit different environmental conditions (Rong et al., 2020).

By revisiting some of the concepts and theories applied in particle interactions and molecular structures formation, we create a minimalist model suitable for robot swarm control. This chapter extends our methodology, presenting a method that allows a swarm of heterogeneous robots to emerge with interesting patterns relying entirely on local interactions with neighbors. Our method consists of modeling the robot swarm as a dynamic GRF and defining the neighborhood system inspired by the Octet rule used in chemistry. By setting the GRF's potential energy as a combination of Coulomb-Buckingham potential and kinetic energy, we formulate a probability distribution used by each robot to sample the most likely velocity commands. As a result, the robots can safely navigate through a bounded environment and bind with others forming dynamic global patterns using only local interactions.

We believe that this method has potential use in various scenarios, especially those where one may want to build more complex structures from simple ones. A possible application is modular robotics, in which complex robots are built from simpler modules and can dynamically change their shape/structure. The bonding behaviors would guide the connection of the modules in a simple and dynamic fashion. Another application is the construction of temporary structures such as bridges and platforms, with different industry and military uses. Figure 7.1 shows an example of a swarm self-organizing to form a bridge in the environment by using robots that mimic atoms of *carbon* and *oxygen*.

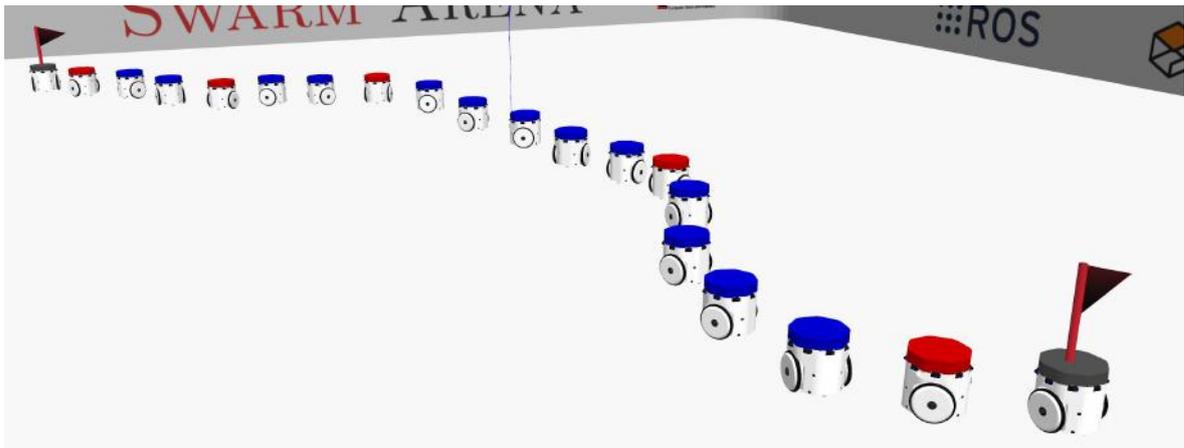


Figure 7.1: Robots mimicking atoms of *carbon* (red) and *oxygen* (blue) create emergent chain patterns useful for dynamic bridge-building applications. Robots with red flags indicate the beginning and end of the chain.

## 7.2 Related Work

Pattern formation occurs in nature at all scales and is a fundamental question across interdisciplinary research, including topics on physical chemistry (Cazalilla et al., 2018; Zhang et al., 2020), cosmology (Liddle and Lyth, 2000), and biochemistry (Maini et al., 1997; Kai et al., 2019). Several patterns or shape formation approaches have also been proposed in the literature for multi-robot systems (Bahceci et al., 2003; Varghese and McKee, 2009a; Liu et al., 2016; Oh et al., 2017; Chennareddy et al., 2017). Most of these works assume global information, which allows each robot in the swarm to directly perceive every other robot (Egerstedt and Hu, 2001; Belta and Kumar, 2002; Pereira and Hsu, 2008; Vickery and Salehi, 2021). Such assumptions allow for fast and efficient convergence of the swarm in pattern formation but may be unrealistic in real applications. Others assume that global information is not always available and deal with the task allocation problem in which robots must coordinate to reach different predefined positions that form patterns or shapes (Hsieh et al., 2008; Varghese and McKee, 2009b; Rahmani et al., 2009; Alonso-Mora et al., 2011; Wang and Rubenstein, 2020). A typical strategy to avoid the requirement of defining positions describing the patterns consists of using seed robots in which some robots do not move and act as a reference to the others helping the swarm to create complex global patterns (Grushin and Reggia, 2010; Rubenstein et al.,

2014). Differently from these, our method assumes minimal and local-only information to produce interesting patterns that resemble molecular shapes. Minimalist approaches are attractive for swarm robotics due to the low sensing capability of the robots, and recent works has shown their feasibility for self-organization problems (Gauci et al., 2014; St-Onge et al., 2018; Lavergne et al., 2019; Mitrano et al., 2019). In the remaining of this section, we discuss some works that tackle the problem using mainly local information and then present the main contributions of our method.

Sahin et al. (2002) designed a robotic system called swarm-bots. The robots can connect to or disconnect from each other using a grasping mechanism enabling self-assemble into different kinds of structures. Inspired from social insect studies (Camazine et al., 2020), the authors employed a probabilistic approach to control the robots. Preliminary results in simulation show that the robot can create patterns, such as a single stripe pattern, beyond the perceptions of individual robots. Despite having a complex dynamic, some aspects of the resulting patterns, such as the mean length of chains, can be controlled through parameters such as the disconnection probability in chain formation behavior. Although the authors do not demonstrate the pattern formation using real robots, the robotic system and control strategy favored the development of several other studies, such as aggregation (Bahgeçi and Sahin, 2005) and self-assembly (Dorigo et al., 2004).

Slavkov et al. (2018) proposed a morphogenesis approach inspired by spontaneous phenomena observed in some biological systems during embryogenesis (Economou et al., 2012). The shapes emerge in a fully self-organized way. The robots rely only on local interactions with neighbors and do not require maps, coordinate systems, or pre-programmed seed robots. The approach uses the concept of robot migration (in analogy with natural developmental biology) and gene regulatory networks (GRN) to create a self-organizing Turing process for pattern formation. The authors successfully demonstrated their method in a swarm of 300 real robots, showing robustness and adaptability in forming Turing patterns.

Further, Carrillo-Zapata et al. (2019) extended the previous approach to increase

the controllability of the system, enabling the formation of specific patterns. The author designed a morphogenesis algorithm based on local gradients for swarms of simple and noisy robots capable of communicating among them. By setting three parameters, robots self-organize to grow controllable shapes while maintaining the communication network. Results demonstrated that the swarm emerges with the rich morphospace of quantitatively different shapes by changing these parameters.

[Li et al. \(2019\)](#) proposed a case study of pattern formation that can be applied to any shape described as a 2D point cloud. The authors present an algorithm that transforms a given point cloud into an acyclic directed graph shared among the swarm members. This graph is used by the control law to allow the swarm to progressively form the target shape based only on local decisions. This means that free robots find their location based on the perceived location of the robots already in the formation. Extensive simulations and experiments on real robots show the effects of swarm size. Results indicate that the algorithm is robust to noise and can handle different formations and shapes.

[Coppola et al. \(2019\)](#) presented a minimalistic approach to generate a local behavior that allows a swarm of homogeneous robots to self-organize into a desired global pattern by relying only on the relative location of their closest neighbors. The generated local behavior is a probabilistic local state-action map, and robots follow policies to select appropriate actions based on their current perception of their neighborhood. Simulations showed the method's robustness against sensor noise and demonstrated the formation of patterns using micro air vehicles. In addition, the authors discuss the scalability of the method and synchronization issues between robots. Although the method uses robots with limited sensory apparatus, it requires a connected topology in the initial configuration of the swarm and an environment discretized by lattices to guarantee the convergence of the swarm in the pattern without suffering from deadlocks.

Unlike other works, we took inspiration from the Octet rule in chemistry to generate patterns with a robotic swarm. Basically, the Octet rule defines the number of bonds each atom preferably makes. The application of the Octet rule in the robotics's context

is yet restricted. [Shiu et al. \(2010\)](#) presented the design of modular robots that uses the Octet rule to dictate attraction force and motion capability. [Randall et al. \(2016\)](#) proposed a decentralized mechanism that aims to simulate chemical reactions using a swarm of miniature robots. The motivation for this work lies in the development of an educational tool to simulate chemical reactions capturing either behavioral or embodied aspects, differently from other tools such as computer simulations or ball-and-stick models. The proposed mechanism replicates what would be expected by simulations of physical-chemical models as faithfully as possible. It uses various built-in sensors to detect neighboring robots and dictate the bonding rules by using direct communication between robots that periodically broadcast state messages.

Unlike these works, our method takes chemistry and particle physics concepts as inspiration to design a probability distribution function that works for controlling robotic swarms. In a nutshell, our method applies the methodology outlined in Chapter 4 to model the swarm as a dynamic GRF and constrains the neighborhood system by the Octet rule. By setting the GRF's potential energy as a combination of Coulomb-Buckingham potential and kinetic energy, we formulate a probability function that indicates in a decentralized fashion which velocities are most likely for each robot given its local perception of the environment. As a result, the swarm creates diverse global patterns that cohesively navigate and adapt to the environment. The contribution of our method comes in many folds. To the best of our knowledge, this is the first work to combine these concepts into swarm robotics bringing another perspective in comparison to other swarm controllers. Our approach is minimalist and multi-emergent, allowing different swarm behaviors to arise indirectly by sampling velocities commands in a probability function. Moreover, it does not rely on global information, goal assignments, communication topology, or preprogrammed seeds to produce patterns.

## 7.3 Method

The proposed method extends the method outlined in Chapter 5, in which we designed and evaluated a novel approach that allowed a swarm of heterogeneous robots to achieve flocking and segregative behaviors simultaneously. After further exploring and improving this method, we realized that by incorporating dynamic constraints to the neighborhood system and adequately defining the swarm heterogeneity and GRF's potentials, the swarm would be able to produce specific patterns, a more complex and restrictive task in comparison to flocking, and with more tangible applications. As mentioned, by taking inspiration from the Octet rule in chemistry, we propose a computational-efficient mechanism to constrain the neighborhood system used by the GRF, enabling the formation of patterns that resemble molecular structures. A detailed description of our method is given in the subsequent sections.

### 7.3.1 Formalization

Building upon the principles detailed in Chapter 4, we draw attention to definition (4.5). This definition underscores that the neighborhood system is restricted only by the sensing range  $\lambda$  and does not differentiate robots of different types. By considering electrical charge as heterogeneity parameters, one may incorporate constraints that allow robots to experience different levels of interaction with their neighbors.

Inspired by some concepts and models of atom interactions in chemistry, we propose a mechanism to restrict a robot to only bind with a certain number of other robots of specific types within its neighborhood. This mechanism is motivated by the *Octet rule* (Gillespie and Silvi, 2002), a relatively simple rule which uses an electron counting formalism for predicting bonding. Based on the Octet theory, the Octet rule generally states that atoms tend to combine so that each of them has eight electrons in their valence shell, as exemplified in Figure 7.2. The principle is that molecules tend to be more stable

when the outer electron shell of each of their atoms is filled with eight electrons. In fact, in nature, all systems tend to acquire as much stability as possible. For example, atoms bind together to form molecules to increase their stability.

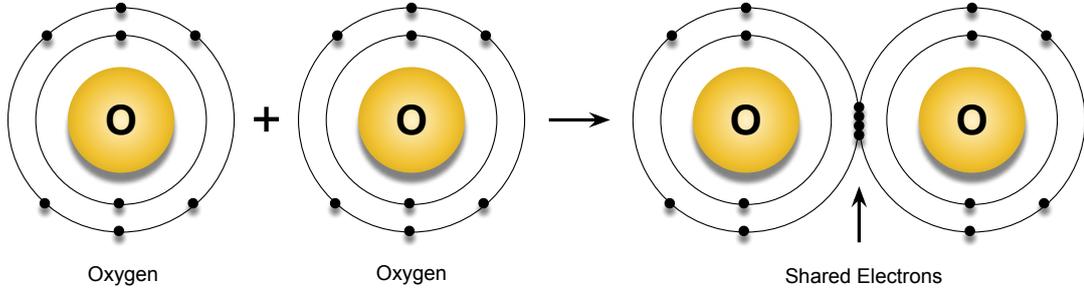


Figure 7.2: The Octet rule refers to the tendency of atoms to prefer to have eight electrons in the valence shell, as exemplified by the pairing of oxygen.

In the context of heterogeneous swarm robots, the proposed mechanism inspired by the *Octet rule* serves as a constraint dictating the interactions among robots. This principle guides robots in determining their proximity to specific types, or distancing themselves from others, based on their individual states and the composition of their immediate neighborhood.

From a mathematical point of view, it is possible to construct such a mechanism by organizing the robots based on their electrical charges and relative distances within a two-dimensional data structure. Formally, we introduce the concept of an ordered neighborhood with respect to the  $i$ -th robot.

**Definition 7.1** (Ordered neighborhood). *The ordered neighborhood for the  $i$ -th robot concerning the Euclidean distance is defined as an ordered set  $\bar{\mathcal{N}}_i \triangleq (\mathcal{N}_i, \preceq)$ , where  $\forall (j, k) \in \bar{\mathcal{N}}_i \rightarrow \|\mathbf{p}_{ij}\| \leq \|\mathbf{p}_{ik}\|$  and  $(j, k)$  are ordered pairs.*

Once the neighbors of the  $i$ -th robot are ordered by distance, let us define the concept of bond partition used to group robots of the same type.

**Definition 7.2** (Bond partition). *The bond partition for the  $i$ -th robot is an ordered partition  $\bar{\mathcal{B}}_i \triangleq (\mathcal{B}_{i,1}, \dots, \mathcal{B}_{i,u}; \succeq)$ , where the ordered set  $(\mathcal{B}_{i,p})$  is a ordered neighborhood containing robots that have the same electrical charge among them,  $\forall (j, k) \in (\mathcal{B}_{i,p}) \rightarrow |c_j| = |c_k|$ . Robots with higher charges have precedence over the ones with lower charges,*

**Algorithm 2** Generating  $\mathcal{B}_i$  for  $i$ -th robot

---

```

1: procedure BONDPARTITION( $\mathcal{N}_i$ )
2:    $\mathcal{B}_i \leftarrow \emptyset$  ▷ Creating bond partition
3:    $\bar{\mathcal{N}}_i \leftarrow \text{sort}(\mathcal{N}_i)$  ▷ Sorting neighbors by distance
4:   for  $j \in \bar{\mathcal{N}}_i$  do
5:     if  $|\mathcal{B}_{j,c_i}| < \mathcal{B}_{j,c_i}^{\max}$  then ▷ Add robot  $j$ -th robot
6:       if  $|\mathcal{B}_{i,c_j}| < \mathcal{B}_{i,c_j}^{\max}$  then
7:          $\mathcal{B}_{i,c_j} \leftarrow \mathcal{B}_{i,c_j} + j$ 
8:    $\bar{\mathcal{B}}_i \leftarrow \text{sort}(\mathcal{B}_i)$  ▷ Sorting partition by charge value
9:   bonds  $\leftarrow 0$  ▷ Count the number of bond
10:  for  $c_k \in \bar{\mathcal{B}}_i$  do
11:    for  $j \in \mathcal{B}_{i,c_k}$  do
12:      if bonds  $< \mathcal{B}_i^{\max}$  then
13:        bonds  $\leftarrow$  bonds + 1 ▷ Add bond
14:      else
15:         $\mathcal{B}_{i,c_k} \leftarrow \mathcal{B}_{i,c_k} - j$  ▷ Remove  $j$ -th robot
16:  return  $\bar{\mathcal{B}}_i$ 

```

---

*i.e.*  $\forall j \in (\mathcal{B}_{i,p})$  and  $\forall k \in (\mathcal{B}_{i,p+1}) \rightarrow |c_j| > |c_k|$ , where the pair  $(p, p + 1)$  define two consecutive ordered sets in  $\bar{\mathcal{B}}_i$ .

From the later definition, we can establish a data structure that allows ordering the robots within the neighborhood both by their distance and their electrical charge. Now, let us state constraints restricting the number of neighbors the robot can bind.

**Definition 7.3** (Maximum bond constraints). *The bond partition  $\bar{\mathcal{B}}_i$  for  $i$ -th robot has a limited number of robots defined by  $\mathcal{B}_i^{\max}$ ,  $|\bar{\mathcal{B}}_i| \leq \mathcal{B}_i^{\max}$ , and robots of the same charge have the same limit,  $\forall (i, j) \in \tau_k : c_i = c_j \rightarrow \mathcal{B}_i^{\max} \triangleq \mathcal{B}_j^{\max}$ . Moreover, the number of robots in each subset of  $\bar{\mathcal{B}}_i$  is also restricted by  $|\bar{\mathcal{B}}_{i,p}| \leq \mathcal{B}_{i,p}^{\max}$ .*

After defining the bond partition, we create a procedure that reduces neighboring robots to such data structure. The mechanism is presented in Algorithm (2). In general, the mechanism prioritizes bonds with closer robots with higher electrical charges, respecting the maximum bond constraints.

**Remark.** *The swarm generates different patterns by setting different maximum bond constraints  $\mathcal{B}_{i,p}^{\max}$  for each  $\mathcal{B}_{i,p} \subset \mathcal{B}_i$  and respecting the previous definitions.*



tential and the kinetic energy as described in Section 5.3 and formulated as,

$$\begin{aligned}
 H_i(\mathbf{v}_i^{(t+1)}, \bar{\mathcal{B}}_i^{(t)}, \mathcal{W}_i^{(t)}) = & \sum_{\forall j \in \mathcal{W}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathbf{w}_j^{(t)}\|) + \\
 & \sum_{\forall j \in \bar{\mathcal{B}}_i^{(t)}} \Phi(\|\mathcal{K}(\mathbf{p}_i^{(t)}, \mathbf{v}_i^{(t+1)}) - \mathcal{K}(\mathbf{p}_j^{(t)}, \mathbf{v}_j^{(t)})\|) + \mathbf{E}_k(\mathbf{V}_i^{(t)}).
 \end{aligned}
 \tag{7.1}$$

Finally, the conditional probability function in Equation (4.6) combined with Equation (7.1) can be computed using Algorithm (1), as detailed in Chapter 4. This algorithm gives the most likely velocity commands for the  $i$ -th robot. With continuous computation, the swarm will converge towards distinct pattern formations influenced by the bond partition parameters.

## 7.4 Experiments and Results

This section demonstrates how a swarm of heterogeneous robots can create diverse patterns using different neighborhood constraints. Further on, we use a realistic simulator as well as real-robot experiments to show that our method may serve as basis for more tangible applications, for example the construction of chain/bridge like structures. The source code and videos of each experiment are available online<sup>1</sup>.

### 7.4.1 Diversity of Patterns

To evaluate the versatility and efficacy of our method in producing different types of patterns with a swarm of robots, we show through numerical simulations four examples where each one has different maximum bond constraints. In all examples, we assume a heterogeneous swarm of  $\eta = 180$  robots, uniformly distributed in  $10 \times 10$  meters bounded environments. The robots are driven by a holonomic kinematic model, reaching a maximum speed of  $v_{max} = 1.0$  meters per second and have a maximum sensing range of

<sup>1</sup>Website: <https://rezeck.github.io/chemistry-inspired-swarm>.

$\lambda = 0.5$  meters. We performed 100 runs with a maximum of 20000 iterations and analyzed the consensus speed among robots in the same group and the persistence of the patterns. As a metric for the persistence, we compute the number of remaining bonds and the number of molecules formed by the swarm over time. Here we consider a molecule as a group of robots that bonds together and has no remaining bonds.

In the first example, we assume a swarm composed of  $|\tau| = 2$  types of robots:  $|\tau_0| = 120$  *hydrogen*-like and  $|\tau_1| = 60$  *oxygen*-like. The mass and electrical charge of the first one is  $m_H = 1$  and  $|c_H| = 1$ , and the second one is  $m_O = 16$  and  $|c_O| = 2$ . We define its bond constraints as  $(\mathcal{B}_{H,O}^{\max}, \mathcal{B}_{H,H}^{\max}) = (1, 1)$  and  $(\mathcal{B}_{O,O}^{\max}, \mathcal{B}_{O,H}^{\max}) = (0, 2)$ , respectively, and each type can have a maximum of  $\mathcal{B}_H^{\max} = 1$  and  $\mathcal{B}_O^{\max} = 2$  bonds allowing the swarm to form a maximum of 60 molecules. Figure 7.4a shows a sequence of snapshots showing the swarm self-organizing to form structures that resemble water molecules. Figure 7.5a shows the mean and the 99% confidence interval for each of the metrics. We note that the average velocity error in the swarm decreases and stabilizes at around  $0.18 \pm 0.02$  meters per second. The mean error oscillations indicate cases where the swarm detects the borders of the environment, requiring the robots to change their velocity to avoid collisions. Regarding the creation and persistence of the patterns, we can see that the swarm converges to the desired patterns. All robots are bonded (only  $0.07 \pm 0.05$  remaining bonds), and form the same number of groups (molecules),  $59.96 \pm 0.09$ , expected for this experiment.

In the second example, we change the *oxygen*-like robots for *carbon*-like robots forming a swarm composed of  $|\tau_0| = 144$  *hydrogen*-like and  $|\tau_1| = 36$  *carbon*-like robots. The mass and electrical charge of the *carbon*-like robots are  $m_C = 12$  and  $|c_C| = 4$ . We define the maximum bond constraints as  $(\mathcal{B}_{H,C}^{\max}, \mathcal{B}_{H,H}^{\max}) = (1, 1)$  and  $(\mathcal{B}_{C,C}^{\max}, \mathcal{B}_{C,H}^{\max}) = (0, 4)$ , respectively, and each type allows a maximum of  $\mathcal{B}_H^{\max} = 1$  and  $\mathcal{B}_C^{\max} = 4$  bonds allowing the swarm to form a maximum of 36 molecules. Figure 7.4b shows the swarm building structures that resemble methane molecules. As seen in the Figure 7.5b, the swarm also reaches consensus in speed, stabilizing at around  $0.21 \pm 0.01$  meters per second, and reduces the number of remaining bonds to  $0.27 \pm 0.36$ . Regarding the number of molecules, we

observed a variation over the iterations but reaching an average of  $35.86 \pm 0.18$  molecules. Unlike the previous pattern, the structures formed are more complex, and as there is a tendency for the molecules to aggregate, some robots may be blocked, making their bond challenging. However, as the method is dynamic, it eventually reaches the number of molecules expected for this experiment.

In the third example, we use  $|\tau| = 3$  types of robots:  $|\tau_0| = 120$  *hydrogen*-like,  $|\tau_1| = 30$  *nitrogen*-like and  $|\tau_2| = 30$  *carbon*-like. The mass and electrical charge of the *nitrogen*-like robots are  $m_N = 14$  and  $|c_N| = 3$ . We define maximum bond constraints as  $(\mathcal{B}_{H,C}^{\max}, \mathcal{B}_{H,N}^{\max}, \mathcal{B}_{H,H}^{\max}) = (1, 1, 1)$ ,  $(\mathcal{B}_{N,C}^{\max}, \mathcal{B}_{N,N}^{\max}, \mathcal{B}_{N,H}^{\max}) = (1, 0, 3)$  and  $(\mathcal{B}_{C,C}^{\max}, \mathcal{B}_{C,N}^{\max}, \mathcal{B}_{C,H}^{\max}) = (2, 1, 2)$ , respectively, and each type can have a maximum of  $\mathcal{B}_H^{\max} = 1$ ,  $\mathcal{B}_N^{\max} = 3$  and  $\mathcal{B}_C^{\max} = 4$  bonds restricting the swarm to form a maximum of 15 molecules. Figure 7.4c shows the swarm forming structures that resemble chemical structure of polyamines. Figure 7.5c shows that the swarm managed to stabilize the average error in velocity (around  $0.19 \pm 0.01$  meters per second) and also reduced the remaining amount of bonds to  $2.20 \pm 0.80$ . As this pattern has three types of robots, different molecules may occur, varying their quantity. However, we also see that this number tends to stabilize near to  $13.96 \pm 0.38$ , closer to the expected value.

In the last example, we assume a swarm composed of  $|\tau_0| = 130$  *oxygen*-like and  $|\tau_1| = 50$  *carbon*-like robots. We define maximum bond constraints as  $(\mathcal{B}_{O,C}^{\max}, \mathcal{B}_{O,O}^{\max}) = (1, 2)$  and  $(\mathcal{B}_{C,C}^{\max}, \mathcal{B}_{C,O}^{\max}) = (0, 2)$ , respectively, and each type can have a maximum of  $\mathcal{B}_O^{\max} = 2$  and  $\mathcal{B}_C^{\max} = 2$  bonds restricting the swarm to form a maximum of 40 molecules. Figure 7.4d shows the swarm creating structures that resemble chemical structures of the oxocarbon. As with the previous patterns, we observed in Figure 7.5d that the swarm achieved consensus in velocity (around  $0.33 \pm 0.01$  meters per second) and reduced the remaining bonds to  $1.37 \pm 0.74$ . We also observed a trend to converge to a specific number of molecules  $16.75 \pm 1.15$ , which is less than the maximum expected value. Unlike the other patterns, the rules allow the formation of larger molecules in the form of chains, reducing the total number of small molecules that the swarm can form.

These experiments show that the swarm is capable of generating different patterns

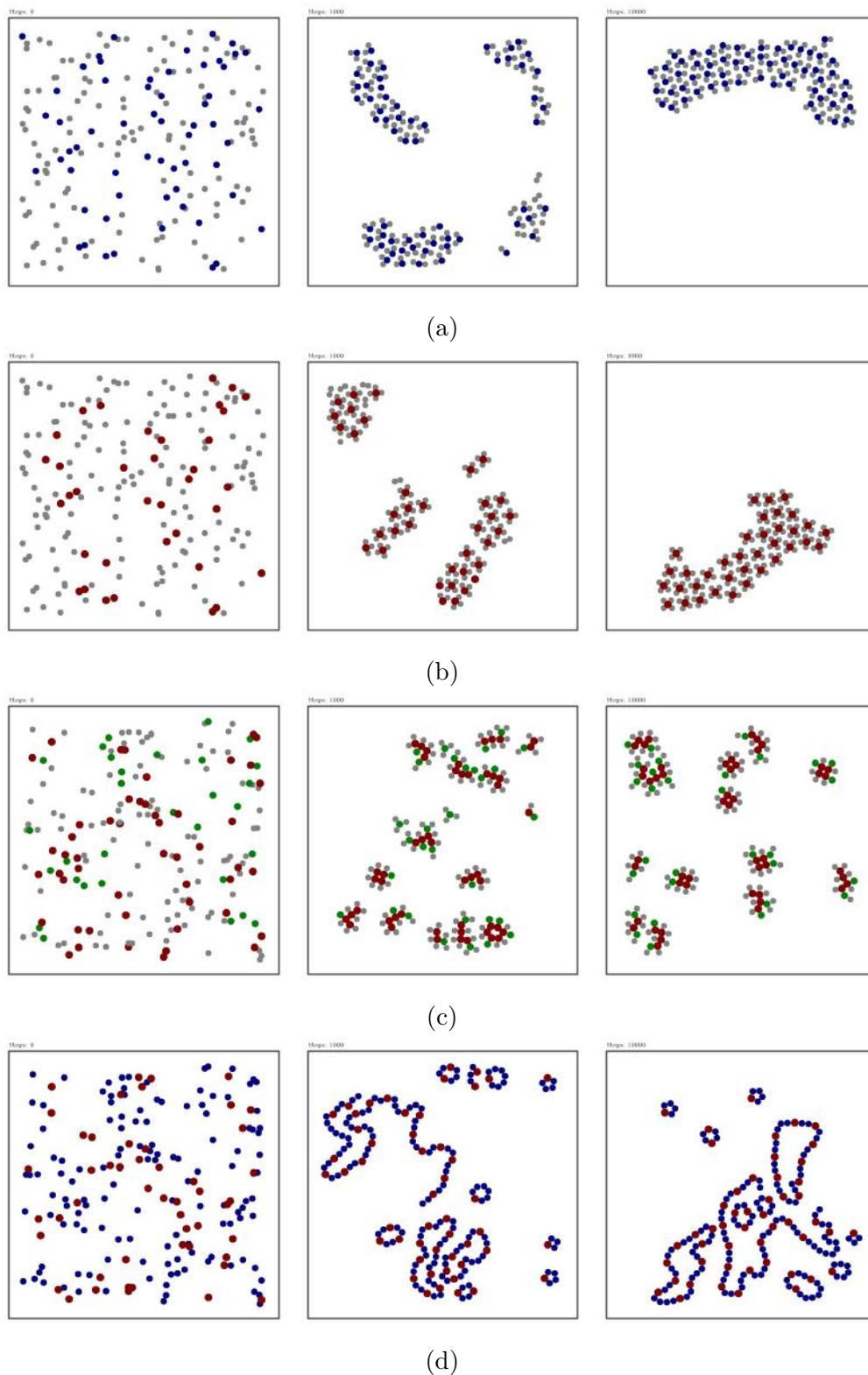


Figure 7.4: Snapshots of four simulated experiments showing that different types of robots create different patterns resembling chemical structures of the (a) water, (b) methane, (c) polyamines and (d) oxocarbons.

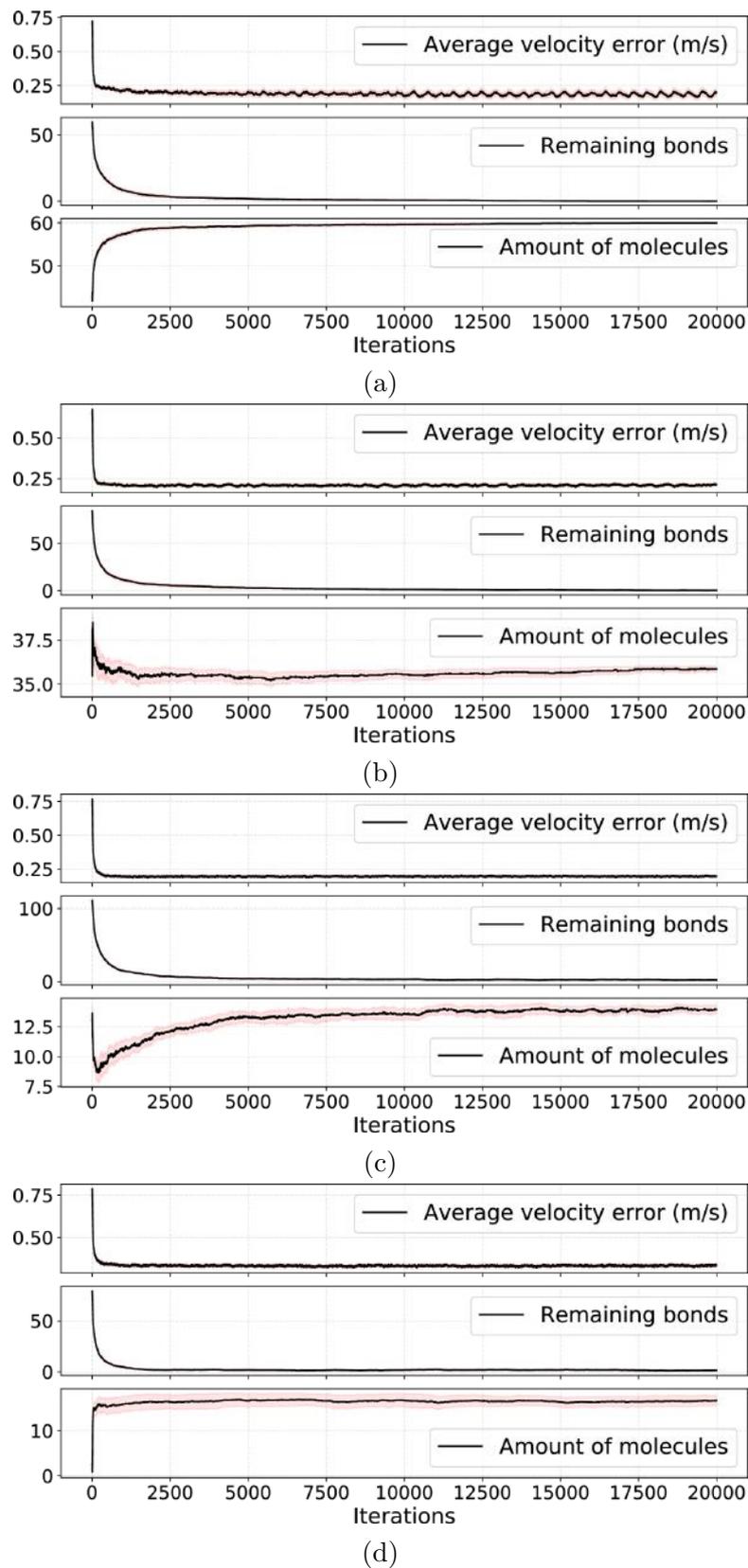


Figure 7.5: Analysis of the persistence and velocity consensus for different patterns: (a) water, (b) methane, (c) polyamines, and (d) oxocarbons. The graphics show the mean and the 99% confidence interval for 100 runs measuring three different metrics. As metrics, we define the average velocity error for each group of robots, the number of remaining bonds, and molecules formed by the swarm in up to 20000 iterations.

and dynamics depending on the binding constraints. In the first two, the swarm produced sub-structures that ended up aggregating. In the third experiment, we observed the formation of more complex and also varied structures. In the last experiment, the swarm created long-chain forms with dynamic behavior that resemble biological systems. These are just a few examples of patterns that our approach can produce.

However, although the method yields a variety of patterns, adjusting its restrictions so that the swarm forms desired structures, such as geometric shapes, is not straightforward without using some form of coordination mechanism. In this sense, in the next section we show how we place robot anchors to build and position chain like structures, which may be the keystone for more concrete applications.

#### 7.4.2 Building Chain-Like Structures with Real Robots

While we take inspiration from chemical reactions so that a swarm can create different shapes in an emergent fashion, an important goal is to use robot swarms for more practical applications. The chain-like oxocarbon structures created by our method are interesting candidates because of the directionality of the emergent patterns. In fact, structure in the form of chains has already been investigated to form paths for foraging problems (Nouyan et al., 2008; Lee et al., 2022). Here we consider employing such behavior to create temporary bridges that could automatically adjust their size and shape to fit any building or terrain.

As a case study, we demonstrate the practical application of the proposed method through physical simulations and real-robot experiments, showcasing its capability to dynamically form structures with a bridge-like topology. For this, in addition to carbon-like and oxygen-like robots, we introduce anchor robots to delimit the ends of the structure, allowing us to control the structure's using other algorithms or human-swarm interaction.

More especially, we assume a swarm composed of  $|\tau_0| = 13$  oxygen-like,  $|\tau_1| = 5$  carbon-like and  $|\tau_2| = 2$  anchor robots. Then, we establish maximum bond constraints

as  $(\mathcal{B}_{O,O}^{\max}, \mathcal{B}_{O,C}^{\max}, \mathcal{B}_{O,A}^{\max}) = (2, 1, 0)$ ,  $(\mathcal{B}_{C,O}^{\max}, \mathcal{B}_{C,C}^{\max}, \mathcal{B}_{C,A}^{\max}) = (2, 0, 1)$  and  $(\mathcal{B}_{A,O}^{\max}, \mathcal{B}_{A,C}^{\max}, \mathcal{B}_{A,A}^{\max}) = (1, 1, 0)$ , respectively. Furthermore, each robot type can establish a maximum of  $\mathcal{B}_O^{\max} = 2$ ,  $\mathcal{B}_C^{\max} = 2$  and  $\mathcal{B}_A^{\max} = 1$  bonds and the anchor robots possess a mass of  $m_A = 200$  and an electrical charge of  $|c_A| = 40$ . This configuration induces the swarm to dynamically create structures, imaging the desired bridge-like topology.

For the experimental validation, we first established a simulated environment within Gazebo, implementing the proposed method through the ROS (Robot Operating System) middleware. Specifically, we simulated twenty HeRo robots, which are compact, cost-effective differential-drive robots we developed, as outlined in Appendix A.

In this experiment, we strategically distributed the robots throughout the environment, positioning two anchor robots at fixed points to serve as the starting and ending anchors for the structure. Figure 7.6 visually depicts the results of the simulation, showcasing the swarm’s ability to converge into a structure that effectively connects the anchor robots. Notably, when moving the anchor robots, the swarm dynamically reconfigures itself to maintain the structure’s integrity.

This observed behavior was successfully replicated in real-world proof-of-concept experiments involving physical HeRo robots, as depicted in Figure 7.7. In this experiment, we deployed five HeRo robots, each remotely controlled through ROS and equipped with locally emulated sensors utilizing images from an overhead camera. To control the robots and ensure compliance with desired velocity vectors, we employed the methodology outlined by [Bruno et al. \(1994\)](#). This approach is particularly adept at converting velocity vectors into linear and angular speeds, suitable for controlling differential robots. In Figure 7.7, the two black blocks function as anchor robots, and the robots seamlessly form a chain-like structure, dynamically adjusting their shape formation in response to manipulations of the anchor positions. The successful replication of this behavior in both simulated and real-world scenarios underscores the efficacy and adaptability of our proposed method in steering swarm formations with practical applications.

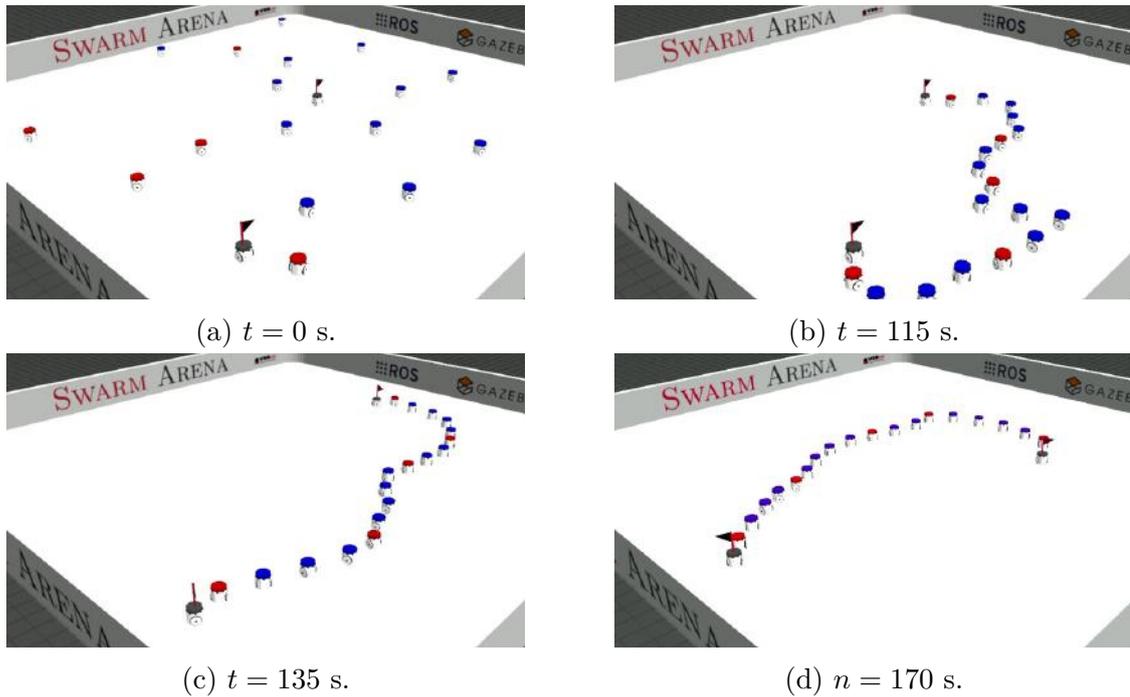


Figure 7.6: 20 robots mimicking atoms of carbon (red) and oxygen (blue) to form shapes with a topology similar to a bridge. Robots with red flags delimit the ends of the structure. In the figure: (a) initial swarm configuration; (b) robots form the bridge; (c) change the position of robots with flag; (d) e (e) change again the position of robots with flag.

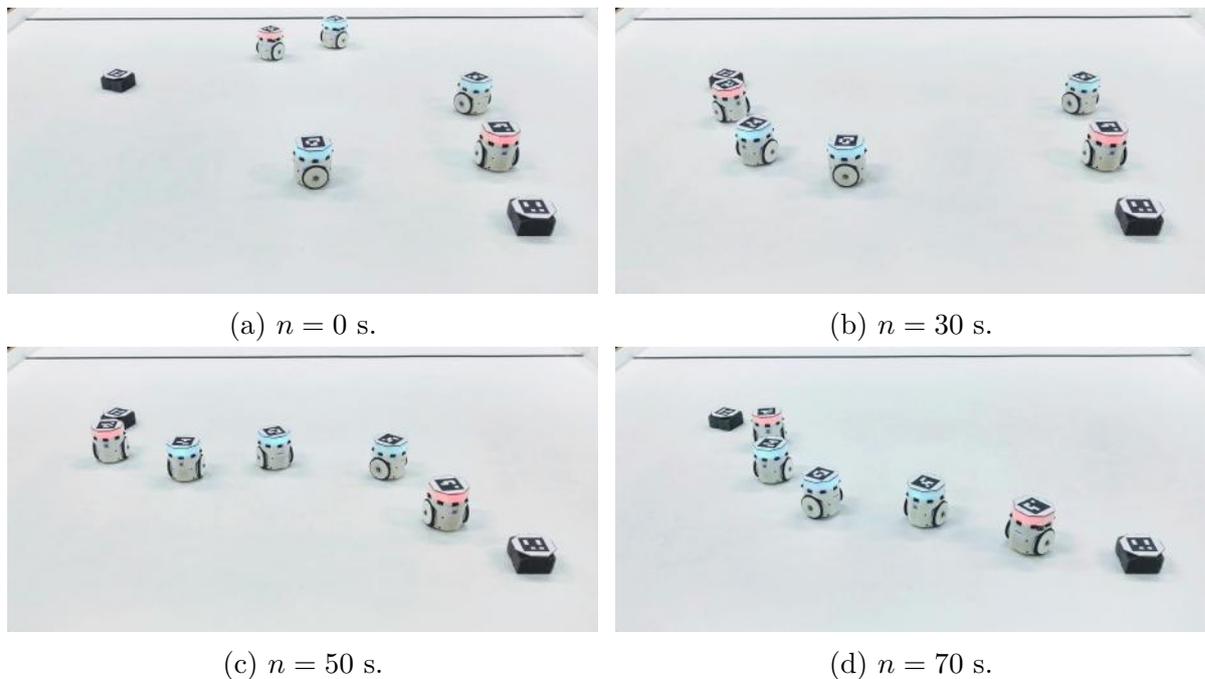


Figure 7.7: Snapshots of an experiment with 5 real robots mimicking 2 carbon (red) and 3 oxygen (blue) atoms creating a dynamic bridge pattern. Black blocks in the image delimit the beginning and end of the structure.

## Chapter 8

# Conclusion and Future Work

In this dissertation, we presented a novel methodology that extends the concepts of Gibbs Random Fields (GRFs) to the context of swarm robotics, allowing us to model and control the behavior of a group of heterogeneous robots. The application of GRFs in swarm robotics is a relatively new and under-explored area but has significant potential for improving the behavior and coordination of robot swarms. Thus, this dissertation aims to contribute to the literature with a novel approach for using GRFs in this field.

### 8.1 Conclusion

The fundamental concept behind using GRFs in swarm robotics consists of modeling the swarm's behavior as a composite of potential functions, where the global minimum of the potential represents the desired behavior or configuration for the swarm. These potential functions encode a conditional probability function that acts as a control mechanism for the robots, guiding them by sampling velocity commands. This induces the robots to move toward the global minimum of the potential, leading the entire swarm to converge to the desired behavior in a decentralized manner. This approach offers several advantages over more traditional methods for modeling and controlling the behavior of a swarm. For instance, it allows for decentralized control, where each robot makes decisions based on local information rather than relying on a central controller. This makes the system more robust and scalable, as there is no single point of failure, and the swarm

can continue to operate even if individual robots fail. Additionally, it allows for a more flexible and adaptable approach to swarm behavior, as the potential function can be easily modified to account for changing environmental conditions or new goals for the swarm.

While these models exhibit significant potential for application in swarm robotics, the correspondence between the potential function and the resulting swarm behavior is not always obvious or direct. It requires a combination of intuition and creativity to design the potential energy function in a way that produces the desired behavior. To address this challenge, we provided a guideline outlining the process of engineering swarm behaviors through the combination of potential functions. Moreover, to illustrate the efficacy of our methodology, we presented the design of methods that tackle three intricate swarm challenges: flocking and segregation, cooperative object transportation, and pattern formation. This serves as a demonstrative showcase of the versatility and applicability of our methodology in tackling diverse swarm robotics behaviors.

The first method described the design of a potential function that encodes aggregation and cohesive navigation behaviors. This enables a swarm of heterogeneous robots to simultaneously exhibit segregative and flocking behaviors. The combination of these behaviors is particularly challenging, especially considering that the robots are initially randomly positioned in the environment, with no guaranteed topology or initial configuration. Within this context, we assumed that each robot is capable of estimating the relative position and velocity of its neighbors, as well as discerning neighbor types to account for heterogeneity and obstacles. To assess the efficacy of this method, we compared it with a deterministic gradient descent-type algorithm utilizing potential differentials. This analysis demonstrates that the latter is easily trapped in local minima. Moreover, we benchmark our segregative behavior against select state-of-the-art approaches and evaluate flocking behavior under the influence of sensor noise. Additionally, we analyzed the method's robustness in complex environments containing diverse obstacles, as well as its resilience in the face of robot failure events. Real-world experiments were also conducted as a proof-of-concept. The thorough results proved that the method adequately segregates a group of heterogeneous robots while simultaneously ensuring cohesive navigation

and obstacle avoidance, even within challenging environments.

The second method focused on the application of our methodology to design a potential energy function that enables robots to navigate through the environment, form groups, locate an object for transportation, and subsequently push it toward a target location. These behaviors emerge through local interactions, eliminating the necessity for explicit communication or centralized coordination. Specifically, the robots are equipped to estimate the relative position and velocity of their neighboring robots, as well as discern between obstacles and objects within their sensing range. Additionally, the robots do not require any specific information about the object, except for its target direction. While there are various technologies capable of providing such information, we assume a scenario where the target location emits a detectable signal, such as light or sound, and the robots are equipped with sensors indicating the direction of this source. To thoroughly evaluate the scalability, adaptability, and robustness of this method, we conducted a series of experiments across various scenarios. The results demonstrate that this method is scalable and capable of supporting the transportation of objects with diverse shapes, sizes, and masses. Furthermore, it exhibited resilience in response to changes in the target location and in the face of robot failure events. Proof-of-concept experiments conducted with real robots further confirmed the method's viability in real-world settings. By implementing a sequence of target directions, we extended the demonstration to showcase the method's application in more complex environments, where the robots effectively maneuvered an object to distinct target locations through a challenging environment.

The third method is built upon the first by setting constraints to the neighborhood system, drawing inspiration from chemical principles that dictate binding polarity between particles. After further exploring the first method, we realized that by incorporating the neighborhood system with dynamic constraints and adequately defining the swarm's heterogeneity alongside the potential functions, the swarm would be able to produce specific patterns. This represents a more intricate and constrained task compared to flocking, with potentially more tangible real-world applications. To evaluate the method's efficacy and capabilities, we conducted experiments showcasing how a swarm of heterogeneous robots

can generate diverse patterns using distinct neighborhood constraints. Furthermore, we utilized both a realistic simulator and real-robot experiments to underscore the potential applications of this method, including tasks such as constructing chain-like structures.

In conclusion, we corroborated through these three methods the flexibility of our methodology in designing mechanisms that adequately produce different swarm behaviors. Overall, it exhibited robustness, adaptability, and scalability across different scenarios, successfully overcame the challenges posed by robot failures and sensor noise, and demonstrated the impact of potential energy function modifications and neighborhood system adjustments on swarm behavior. Furthermore, our system's scalability was underscored through several simulations involving large swarms. Finally, these results established the efficacy and broad applicability of our methodology within the field of swarm robotics, setting the stage for promising future research directions.

## 8.2 Future Work

While this dissertation provides a concise exploration of Gibbs Random Fields (GRFs) applications in swarm robotics, it is important to highlight that our investigations were primarily focused on swarms constrained in various sensing aspects. However, there exist numerous extensions, both in the methodology and its application, that can lead to the generation of diverse swarm behaviors when relaxing the restrictions assumed here.

- One route lies in the method for object transportation. By assuming that robots estimate the object's target position and orientation, it is possible to extend the potential functions with a more sophisticated caging behavior. This allows the robots to move the object more efficiently, controlling its orientation with respect to the target. This could be achieved by adapting the move-around behavior and introducing heterogeneity with two types of robots segregating in the caging behavior. With each group independently pushing the object, its orientation can be influenced.

- Another interesting application involves cooperative coverage, where the swarm must spatially cover a given region in the environment. Combining the move-around behavior with group formation could lead to dynamic circumnavigation of the coverage region. As the swarm size increases, the coverage should be achieved more dynamically and efficiently.
- The field of human-swarm interaction also presents an interesting challenge since it requires the control of multiple robots simultaneously in a decentralized fashion. One approach could involve assuming that a human controls a single robot, and this robot exerts a stronger influence over the others. This robot would indirectly convey the human's intentions to the swarm. Additionally, this concept can be extended to incorporate haptic devices, allowing for the transmission of forces to the human operator. This way, the operator can sense the swarm's interactions with each other, as well as with obstacles and objects in the environment.
- Alongside applications, there are potential extensions to the methodology that could enhance its flexibility for a broader range of use cases. One notable characteristic of the proposed methodology is that the swarm is in continuous motion, which may be inefficient for certain applications. Investigating how to encode stopping conditions that guarantee the swarm reaches the desired behavior, while adhering to a decentralized paradigm, could be a promising research direction. One approach could be setting the temperature parameter in the Gibbs distribution as a function, allowing for the cooling down of the swarm to encourage even lower velocity sampling.
- Moreover, it is equally important to investigate theoretical proofs of convergence when relaxing the assumptions of GRFs framework, particularly with dynamic neighborhoods. Analyzing the convergence properties under dynamic conditions can provide valuable insights into the robustness and stability of our methodology.

Furthermore, by relaxing the constraints assumed in this dissertation, there is considerable potential to extend the methodology's applicability for achieving more complex and efficient swarm behaviors.

# Bibliography

- Muhanad H Mohammed Alkilabi, Aparajit Narayan, and Elio Tuci. Cooperative object transport with a swarm of e-puck robots: robustness and scalability of evolved collective strategies. *Swarm intelligence*, 11(3):185–209, 2017.
- Muhanad H Mohammed Alkilabi, Aparajit Narayan, Chuan Lu, and Elio Tuci. Evolving group transport strategies for e-puck robots: moving objects towards a target area. In *Distributed Autonomous Robotic Systems*, pages 503–516. Springer, 2018.
- J Alonso-Mora, A Breitenmoser, M Rufli, R Siegwart, and P Beardsley. Multi-robot system for artistic pattern formation. In *2011 IEEE Int. Conf. on Robotics and Automation*, pages 4512–4517. IEEE, 2011.
- E Bahceci, O Soysal, and E Sahin. A review: Pattern formation and adaptation in multi-robot systems. *Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-03-43*, 2003.
- E. Bahgeçi and E. Sahin. Evolving aggregation behaviors for swarm robotic systems: A systematic case study. In *Swarm Intelligence Symposium, 2005. SIS 2005.*, pages 333–340. IEEE, 2005.
- John S Baras and Xiaobo Tan. Control of autonomous swarms using gibbs sampling. In *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, volume 5, pages 4752–4757. IEEE, 2004.
- Meysam Basiri, Felix Schill, Dario Floreano, and Pedro U Lima. Audio-based localization for swarms of micro air vehicles. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 4729–4734. IEEE, 2014.

- C Belta and V Kumar. Trajectory design for formations of robots by kinetic energy shaping. In *2002 IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2593–2598. IEEE, 2002.
- Calin Belta and Vijay Kumar. Abstraction and control for groups of robots. *IEEE Transactions on robotics*, 20(5):865–875, 2004.
- Andrew Blake, Pushmeet Kohli, and Carsten Rother. *Markov random fields for vision and image processing*. MIT press, 2011.
- Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.
- Siciliano Bruno, Sciavicco Lorenzo, Villani Luigi, and Oriolo Giuseppe. Robotics: modelling, planning and control, 2010. *Cited on*, 1, 1994.
- Richard A Buckingham. The classical equation of state of gaseous helium, neon and argon. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 168(933):264–283, 1938.
- S Camazine, JL Deneubourg, NR Franks, J Sneyd, G Theraula, and E Bonabeau. *Self-organization in biological systems*. Princeton University press, 2020.
- D Carrillo-Zapata, J Sharpe, AFT Winfield, L Giuggioli, and S Hauert. Toward controllable morphogenesis in large robot swarms. *Robotics and Automation Letters*, 4(4):3386–3393, 2019.
- AL Casalilla, D Chowdhury, A Ilinov, S Mondal, P Barman, SR Bhattacharyya, D Ghose, F Djurabekova, K Nordlund, and S Norris. Pattern formation on ion-irradiated si surface at energies where sputtering is negligible. *Journal of Applied Physics*, 123(23):235108, 2018.
- Jianing Chen, Melvin Gauci, Michael J Price, and Roderich Groß. Segregation in swarms of e-puck robots based on the brazil nut effect. In *Proceedings of the 11th International*

- Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 163–170, 2012.
- Jianing Chen, Melvin Gauci, Wei Li, Andreas Kolling, and Roderich Groß. Occlusion-based cooperative transport with a swarm of miniature mobile robots. *IEEE Transactions on Robotics*, 31(2):307–321, 2015.
- S Chennareddy, A Agrawal, and A Karuppiah. Modular self-reconfigurable robotic systems: a survey on hardware architectures. *Journal of Robotics*, 2017, 2017.
- M Coppola, J Guo, E Gill, and GHE de Croon. Provable self-organizing pattern formation by a swarm of robots with limited knowledge. *Swarm Intelligence*, 13(1):59–94, 2019.
- M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, G. Baldassarre, S. Nolfi, J. Deneubourg, F. Mondada, D. Floreano, et al. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2):223–245, 2004.
- Frederick Ducatelle, Gianni A Di Caro, Carlo Pinciroli, and Luca M Gambardella. Self-organized cooperation between robotic swarms. *Swarm Intelligence*, 5(2):73, 2011.
- Gregory Dudek, Michael RM Jenkin, Evangelos Miliotis, and David Wilkes. A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4):375–397, 1996.
- Russell Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pages 39–43. IEEE, 1995.
- AD Economou, A Ohazama, T Porntaveetus, PT Sharpe, S Kondo, MA Basson, A Gritli-Linde, MT Cobourne, and JBA Green. Periodic stripe formation by a turing mechanism operating at growth zones in the mammalian palate. *Nature genetics*, 44(3):348–351, 2012.
- BF Edson Filho and Luciano CA Pimenta. Segregating multiple groups of heterogeneous units in robot swarms using abstractions. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 401–406. IEEE, 2015.

- M Egerstedt and X Hu. Formation constrained multi-agent control. *IEEE transactions on robotics and automation*, 17(6):947–951, 2001.
- Jan Faigl, Tomáš Krajník, Jan Chudoba, Libor Přeučil, and Martin Saska. Low-cost embedded system for relative localization in robotic swarms. In *2013 IEEE International Conference on Robotics and Automation*, pages 993–998. IEEE, 2013.
- Malintha Fernando. Online flocking control of uavs with mean-field approximation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8977–8983. IEEE, 2021.
- Edson B Ferreira-Filho and Luciano CA Pimenta. Abstraction based approach for segregation in heterogeneous robotic swarms. *Robotics and Autonomous Systems*, 122:103295, 2019.
- Edson B Ferreira Filho and Luciano CA Pimenta. Segregation of heterogeneous swarms of robots in curves. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7173–7179. IEEE, 2020.
- Paolo Fiorini and Zvi Shiller. Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, 17(7):760–772, 1998.
- Ryusuke Fujisawa, Hikaru Imamura, and Fumitoshi Matsuno. Cooperative transportation by swarm robots using pheromone communication. In *Distributed Autonomous Robotic Systems*, pages 559–570. Springer, 2013.
- M Gauci, J Chen, W Li, TJ Dodd, and R Groß. Self-organized aggregation without computation. *The International Journal of Robotics Research*, 33(8):1145–1161, 2014.
- R. J. Gillespie and R. S. Nyholm. Inorganic stereochemistry. *Quarterly Reviews, Chemical Society*, 11(4):339–380, 1957.
- RJ Gillespie and B Silvi. The octet rule and hypervalence: two misunderstood concepts. *Coord. Chemistry Rev.*, 233:53–62, 2002.

- Roderich Groß, Stéphane Magnenat, and Francesco Mondada. Segregation in swarms of mobile robots based on the brazil nut effect. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4349–4356. IEEE, 2009.
- Alexander Grushin and James A Reggia. Parsimonious rule generation for a nature-inspired approach to self-assembly. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 5(3):1–24, 2010.
- W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. ISSN 00063444. URL <http://www.jstor.org/stable/2334940>.
- MA Hsieh, V Kumar, and L Chaimowicz. Decentralized controllers for shape generation with robotic swarms. *Robotica*, 26(5):691–701, 2008.
- Fabrício R Inácio, Douglas G Macharet, and Luiz Chaimowicz. United we move: Decentralized segregated robotic swarm navigation. In *Distributed Autonomous Robotic Systems*, pages 313–326. Springer, 2018.
- Fabrício R Inácio, Douglas G Macharet, and Luiz Chaimowicz. Pso-based strategy for the segregation of heterogeneous robotic swarms. *Journal of Computational Science*, 31:86–94, 2019.
- Devwrat Joshi, Masahiro Shimizu, and Koh Hosoda. Segregation and flow of modules in a robot swarm utilising the brazil nut effect. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4080–4085. IEEE, 2019.
- Y Kai, C Chang, F Wu, W Chen, and C Yeh. Formation mechanism of ni<sub>2</sub>ti<sub>4</sub>ox in niti shape memory alloy. *Materialia*, 5:100194, 2019.
- Gabriel Kapellmann-Zafra, Jianing Chen, and Roderich Groß. Using google glass in human–robot swarm interaction. In *Annual Conference Towards Autonomous Robotic Systems*, pages 196–201. Springer, 2016.

- Y Kim, S Lee, HS Yang, and DA Shell. Toward autonomous robotic containment booms: visual servoing for robust inter-vehicle docking of surface vehicles. *Intelligent Service Robotics*, 5(1):1–18, 2012.
- Ross Kindermann. Markov random fields and their applications. *American mathematical society*, 1980.
- Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
- C Ronald Kube. Task modelling in collective robotics. *Autonomous robots*, 4(1):53–72, 1997.
- C Ronald Kube and Eric Bonabeau. Cooperative transport by ants and robots. *Robotics and autonomous systems*, 30(1-2):85–101, 2000.
- C Ronald Kube and Hong Zhang. Collective robotics: From social insects to robots. *Adaptive behavior*, 2(2):189–218, 1993.
- Manish Kumar, Devendra P Garg, and Vijay Kumar. Segregation of heterogeneous units in a swarm of robotic agents. *IEEE transactions on automatic control*, 55(3):743–748, 2010.
- FA Lavergne, H Wendehenne, T B auerle, and C Bechinger. Group formation and cohesion of active particles with visual perception–dependent motility. *Science*, 364(6435):70–74, 2019.
- Dohee Lee, Qi Lu, and Tsz-Chiu Au. Dynamic robot chain networks for swarm foraging. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4965–4971, 2022. doi: 10.1109/ICRA46639.2022.9811625.
- G Li, D St-Onge, C Pinciroli, A Gasparri, E Garone, and G Beltrame. Decentralized progressive shape formation with robot swarms. *Autonomous Robots*, 43(6):1505–1521, 2019.

- A. R. Liddle and D. H. Lyth. *Cosmological inflation and large-scale structure*. Cambridge university press, 2000.
- J Liu, X Zhang, and G Hao. Survey on research and development of reconfigurable modular robots. *Advances in Mechanical Engineering*, 8(8):1687814016659597, 2016.
- PK Maini, KJ Painter, and HNP Chau. Spatial pattern formation in chemical and biological systems. *Journal of the Chemical Society, Faraday Transactions*, 93(20):3601–3610, 1997.
- Peter Mitrano, Jordan Burklund, Michael Giancola, and Carlo Pinciroli. A minimalistic approach to segregation in robot swarms. In *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 105–111. IEEE, 2019.
- Sifat Momen, Bala P Amavasai, and Nazmul H Siddique. Mixed species flocking for heterogeneous robotic swarms. In *EUROCON 2007-The International Conference on "Computer as a Tool"*, pages 2329–2336. IEEE, 2007.
- Francesco Mondada, Michael Bonani, Xavier Raemy, James Pugh, Christopher Cianci, Adam Klaptocz, Stephane Magnenat, Jean-Christophe Zufferey, Dario Floreano, and Alcherio Martinoli. The e-puck, a robot designed for education in engineering. In *Proceedings of the 9th conference on autonomous robot systems and competitions*, volume 1, pages 59–65. IPCB: Instituto Politécnico de Castelo Branco, 2009.
- Mikko Mönkkönen, Pekka Helle, and Kimmo Soppela. Numerical and behavioural responses of migrant passerines to experimental manipulation of resident tits (*parus spp.*): heterospecific attraction in northern breeding bird communities? *Oecologia*, 85(2):218–225, 1990.
- Shervin Nouyan, Alexandre Campo, and Marco Dorigo. Path formation in a robot swarm. *Swarm Intelligence*, 2(1):1–23, 2008.
- H Oh, AR Shirazi, C Sun, and Y Jin. Bio-inspired self-organising multi-robot pattern formation: A review. *Robotics and Autonomous Systems*, 91:83–100, 2017.

- AR Pereira and L Hsu. Adaptive formation control using artificial potentials for euler-lagrange agents. *IFAC*, 41(2):10788–10793, 2008.
- A Rahmani, M Ji, M Mesbahi, and M Egerstedt. Controllability of multi-agent systems from a graph-theoretic perspective. *SIAM Journal on Control and Optimization*, 48(1):162–186, 2009.
- A Randall, J Klingner, and N Correll. Simulating chemical reactions using a swarm of miniature robots. In *International Conference on Simulation of Adaptive Behavior*, pages 305–316. Springer, 2016.
- Craig W Reynolds. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 25–34, 1987.
- Paulo Rezek. Hero: An open platform for robotics research and education. Master’s thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG - Brazil, 2019.
- Paulo Rezek, Héctor Azpúrua, Maurício F. S. Corrêa, and Luiz Chaimowicz. Hero 2.0: a low-cost robot for swarm robotics research. *Autonomous Robots*, 2023. doi: 10.1007/s10514-023-10100-0. URL <https://doi.org/10.1007/s10514-023-10100-0>.
- Steven Roelofsen, Denis Gillet, and Alcherio Martinoli. Reciprocal collision avoidance for quadrotors using on-board visual detection. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4810–4817. IEEE, 2015.
- H. H. Rong, W. Tu, F. Duarte, and C. Ratti. Employing waterborne autonomous vehicles for museum visits: A case study in amsterdam. *European Transport Research Review*, 12(1):1–13, 2020.
- M Rubenstein, A Cornejo, and R Nagpal. Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799, 2014.
- E Sahin, TH Labella, V Trianni, JL Deneubourg, P Rasse, D Floreano, L Gambardella, F Mondada, S Nolfi, and M Dorigo. Swarm-bot: Pattern formation in a swarm of self-

- assembling mobile robots. In *IC on Systems, Man and Cybernetics*, volume 4, pages 6–pp. IEEE, 2002.
- Erol Sahin and Alan FT Winfield. Special issue on swarm robotics. *Swarm Intell.*, 2(2-4): 69–72, 2008.
- D. Saintillan and M. J. Shelley. Instabilities and pattern formation in active particle suspensions: kinetic theory and continuum simulations. *Physical Review Letters*, 100(17):178103, 2008.
- Vinicius G. Santos, Luciano C. A. Pimenta, and Luiz Chaimowicz. Segregation of multiple heterogeneous units in a robotic swarm. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1112–1117. IEEE, 2014a.
- Vinicius G. Santos, Anderson G. Pires, Reza J. Alitappeh, Paulo A. F Rezeck, Luciano C. A. Pimenta, Douglas G. Macharet, and Luiz Chaimowicz. Spatial segregative behaviors in robotic swarms using differential potentials. *Swarm Intelligence*, pages 1–26, 2020.
- Vinicius Graciano Santos, Mario FM Campos, and Luiz Chaimowicz. On segregative behaviors using flocking and velocity obstacles. In *Distributed Autonomous Robotic Systems*, pages 121–133. Springer, 2014b.
- Melanie Schranz, Martina Umlauft, Micha Sende, and Wilfried Elmenreich. Swarm robotic behaviors and current applications. *Frontiers in Robotics and AI*, 7:36, 2020.
- M Shah, SK Shah, and M Shah. Autonomous robotic vehicle for oil spills cleaning with nano particles. In *2018 Int. Conf. on Manipulation, Automation and Robotics at Small Scales*, pages 1–6. IEEE, 2018.
- C Shiu, C Fu, T Lee, and L Lian. Modular design of a reconfigurable electromagnetic robot. *Advanced Robotics*, 24(7):1059–1078, 2010.
- I Slavkov, D Carrillo-Zapata, N Carranza, X Diego, F Jansson, J Kaandorp, S Hauert, and J Sharpe. Morphogenesis in robot swarms. *Science Robotics*, 3(25), 2018.

- David St-Onge, Carlo Pinciroli, and Giovanni Beltrame. Circle formation with computation-free robots shows emergent behavioural structure. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5344–5349. IEEE, 2018.
- Malcolm S Steinberg. Reconstruction of tissues by dissociated cells. *Science*, 141(3579): 401–408, 1963.
- Alessandro Stranieri, Eliseo Ferrante, Ali Emre Turgut, Vito Trianni, Carlo Pinciroli, Mauro Birattari, and Marco Dorigo. Self-organized flocking with an heterogeneous mobile robot swarm. In *ECAL*, pages 789–796, 2011.
- Xiaobo Tan, Wei Xi, and John S Baras. Decentralized coordination of autonomous swarms using parallel gibbs sampling. *Automatica*, 46(12):2068–2076, 2010.
- Ying Tan and Zhong-yang Zheng. Research advance in swarm robotics. *Defence Technology*, 9(1):18–39, 2013.
- Danesh Tarapore, Roderich Groß, and Klaus-Peter Zauner. Sparse robot swarms: Moving swarms to real-world applications. *Front. Robot. AI* 7: 83. doi: 10.3389/frobt, 2020.
- Elio Tuci, Muhanad HM Alkilabi, and Otar Akanyeti. Cooperative object transport in multi-robot systems: A review of the state-of-the-art. *Frontiers in Robotics and AI*, 5: 59, 2018.
- Jur van den Berg, Stephen J. Guy, Ming Lin, and Dinesh Manocha. *Reciprocal n-Body Collision Avoidance*, pages 3–19. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-19457-3. doi: 10.1007/978-3-642-19457-3\_1. URL [http://dx.doi.org/10.1007/978-3-642-19457-3\\_1](http://dx.doi.org/10.1007/978-3-642-19457-3_1).
- Jur Van Den Berg, Stephen J Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *Robotics research*, pages 3–19. Springer, 2011.
- Erik Vanmarcke. *Random fields: analysis and synthesis*. World scientific, 2010.

- B Varghese and G McKee. A review and implementation of swarm pattern formation and transformation models. *International J. of Intelligent Computing and Cybernetics*, 2009a.
- B Varghese and G McKee. Towards a unifying framework for pattern transformation in swarm systems. In *AIP Conf. Proc.*, volume 1107, pages 65–70. American Institute of Physics, 2009b.
- E. J. W. Verwey. Theory of the stability of lyophobic colloids. *The Journal of Physical Chemistry*, 51(3):631–636, 1947.
- C Vickery and SA Salehi. A mean shift-based pattern formation algorithm for robot swarms. In *2021 7th Int. Conf. on Automation, Robotics and Applications (ICARA)*, pages 16–20. IEEE, 2021.
- J. H. Von Brecht, D. Uminsky, T. Kolokolnikov, and A. L. Bertozzi. Predicting pattern formation in particle interactions. *Mathematical Models and Methods in Applied Sciences*, 22(supp01):1140002, 2012.
- H Wang and M Rubenstein. Shape formation in homogeneous swarms using local task swapping. *Trans. on Robotics*, 36(3):597–612, 2020.
- John Wang and Edwin Olson. Apriltag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4193–4198. IEEE, 2016.
- Gerhard Winkler. *Image analysis, random fields and Markov chain Monte Carlo methods: a mathematical introduction*, volume 27. Springer Science & Business Media, 2003.
- Wei Xi and John S Baras. Gibbs sampler based control of autonomous vehicle swarms in the presence of sensor errors. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 5084–5090. IEEE, 2006.
- Wei Xi, Xiaobo Tan, and John S Baras. Gibbs sampler-based coordination of autonomous swarms. *Automatica*, 42(7):1107–1119, 2006.

---

Seiji Yamada and Jun'ya Saito. Adaptive action selection without explicit communication for multirobot box-pushing. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 31(3):398–404, 2001.

Z Zhang, K Yao, L Feng, J Hu, and C Chin. Pattern formation in a driven bose-einstein condensate. *Nat. Physics*, 16(6):652–656, 2020.

# Appendix A

## HeRo 2.0: a Low-Cost Robot for Swarm Robotics Research

In this dissertation, we present a novel methodology for generating different types of swarm behaviors in a decentralized way using only local sensing. This is a significant contribution to the field of swarm robotics, as it allows for creating complex swarm behaviors without the need for centralized control or any global information. In addition, this work also includes significant contributions in improving the design and control of a low-cost swarm robotic platform called HeRo, originally presented as the main contribution of this author's master's thesis (Rezeck, 2019). This additional work was fundamental for building a group of robots able to perform real-world experiments that could demonstrate the methodology's effectiveness. In the following, we briefly present some of these improvements describing the main features and capabilities of this new version of the robot, HeRo 2.0 (Rezeck et al., 2023).

To build a swarm-capable robotic platform, several requirements need to be considered as objectives. One of the primary objectives is affordability since swarm teams often consist of a large number of robots. Therefore, the robots should be as inexpensive as possible while also being small and equipped with some form of sensing capability to allow interaction with the environment. Additionally, they should have long-term power autonomy since the swarm may need to operate long enough for the collective behavior to emerge. Another important requirement is reliability. The robots should be highly fault-tolerant to avoid any failures that could negatively affect the swarm's performance.

Communication capabilities should also support a large number of robots to ensure scalability, which means that the robots should be able to successfully perform different tasks, even when the number of robots increases. Furthermore, the robots must be easily reproducible, meaning they should be easily assembled and not use hard-to-acquire or extremely hard-to-assemble components. Finally, the robots should be easily programmable and compatible with modern robotic frameworks and development pipelines to ensure that they are easily adaptable to different swarm applications.

It is challenging to satisfy all of these conditions in a single design. The design choices concerning one requirement, such as size, produce additional constraints on others, such as sensing and powering. Therefore, the design process should simultaneously consider all of these constraints and find convenient design solutions that are suitable for multi-purpose applications. By considering the maximum use of commercially available components for ease of production and assembly and minimum possible price without sacrificing the processing power and sensing capabilities, we present the best-suited design for HeRo (Fig. A.1) after evaluating multiple microcontroller boards, wireless technologies, sensors, actuators, and model designs for additive manufacturing.

In this new version, the mainboard uses an Espressif ESP8266 (32-bits 160 MHz) microcontroller to perform the motors' control and acquire and process sensor data. This microcontroller houses a built-in Wi-Fi module, allowing the robots to communicate among themselves robustly and reliably using TCP/IP protocols. The locomotion system uses two differential-driven wheels reaching a maximum speed of 25 cm/s. The board houses a set of sensors, such as 8 IR sensors that provide ambient light measurements and distance estimations with a range of 20 cm, an inertial motion unit for improved odometry and general use, and two rotary encoders for localization and motion control. The mainboard is also modular, allowing users to attach several other components, such as a camera, motors, displays, or transceivers for communication or localization. To facilitate programming, HeRo supports FOTA (Firmware Over-The-Air) using a Wi-Fi interface. Such technology allows users to upload their codes on many robots remotely. Moreover, HeRo is also a ROS-compatible robot and communicates using a TCP/IP connection with

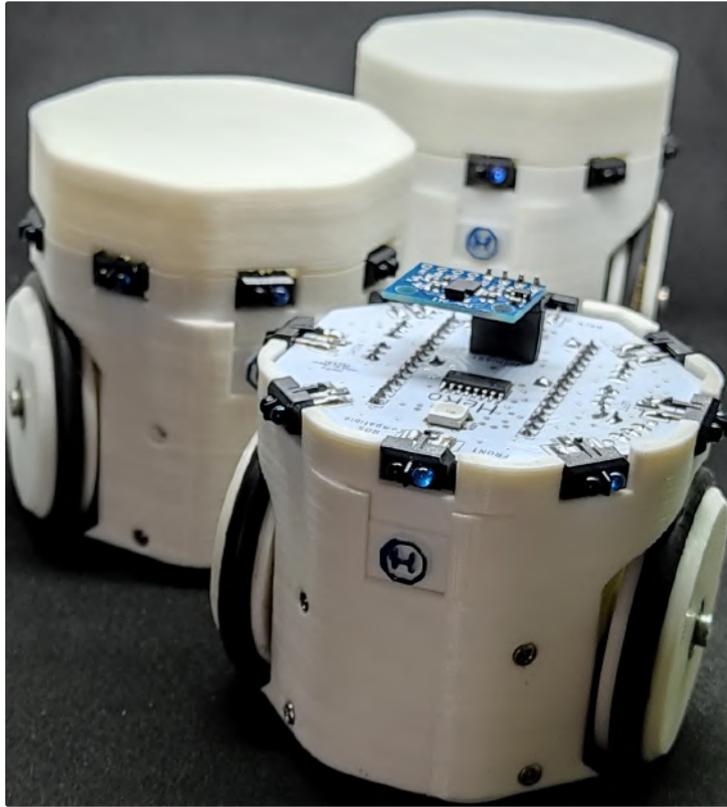


Figure A.1: Design of the open swarm robotic platform. The body of the robots was designed for and fabricated using additive manufacturing.

a remote computer executing ROS. Since the robot's autonomy is an important factor considering the time and number of experiments, HeRo uses a 1800 mA LiPo battery reaching up to 9 hours of use. Finally, it is worth mentioning that HeRo is open hardware and software, allowing for customization and easy sharing of code and designs. It is also provided with a simulated model with a test environment in Gazebo Simulator to help develop and evaluate new swarm algorithms. More details of HeRo 2.0 can be found in [Rezeck et al. \(2023\)](#).