Density-Based Guidance and Control for Decentralized Autonomous Swarms

Utku Eren

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2026

Reading Committee:

Behçet Açıkmeşe, Chair

Mehran Mesbahi

Amir Taghvaei

Program Authorized to Offer Degree:
Aeronautics & Astronautics

University of Washington

Abstract

Density-Based Guidance and Control for Decentralized Autonomous Swarms

Utku Eren

Chair of the Supervisory Committee:
Professor Behçet Açıkmeşe
William E. Boeing Department of Aeronautics & Astronautics

Autonomous swarms are an ensemble of thousands of expendable, ideally low-cost, systems or robots working collaboratively to accomplish missions in harsh environments, where a single monolithic system is either too risky or impractical for successful completion of the collective objective. Due to their distinguishing advantages in redundancy, reconfigurability, and robustness, guidance and control of autonomous swarms have become a rapidly emerging research area, with particularly strong interest in future space missions. This dissertation addresses both the guidance and control problems in a hierarchically decomposed framework in order to keep each level tractable while ensuring mathematical guarantees such as stability and convergence at each level of the hierarchy. As a result, the proposed solutions at each level employ different mathematical tools and methods, while the overall framework is unified through the notion of spatial density.

The first part of the dissertation addresses the guidance level of the proposed hierarchical framework by focusing on high-level density sequence design that interfaces with lower-level swarm control. It presents additional results extending prior work on Markov chain-based synthesis of desired density sequences. New constraints are introduced into the Markov chain synthesis optimization problem to provide additional control over the transitional behavior of target density sequences, particularly with respect to the spatial trajectory probabilities of individual agents.

The second and core part of the dissertation addresses the control level of the hierarchical framework by developing a decentralized, density-based swarm control algorithm for tracking desired spatial density distributions. The proposed control algorithm assumes heat equation-based transition dynamics for the swarm and employs a density feedback control law based on nonparametric local density estimation from local agent measurements. Stability and convergence of the resulting closed-loop system are established in the continuum limit, yielding asymptotic guarantees with increasing numbers of agents. In addition, formal collision avoidance guarantees are derived for a class of kernel functions used in the local density estimation process. Finally, the control framework is extended with an adaptive mechanism that enables decentralized, online optimization of the density estimation parameters, allowing each agent to adjust its estimation strategy in real time using only local information.

TABLE OF CONTENTS

Page

List of Figures iii

List of Tables vi

Chapter 1: Introduction 1
1.1 Research Motivation and Objectives 2
1.2 State Of The Art 5
1.3 Proposed Approach 9
1.4 Thesis Outline 11

Chapter 2: Density Guidance Problem 13
2.1 Background 13
2.2 Problem Formulation 16
2.3 Markov Chain Synthesis Problem with Constraints 20
2.4 Numerical Simulations & Experimental Validation 31

Chapter 3: Density Feedback Control 37
3.1 Background 37
3.2 Problem Formulation 40
3.3 Local Density Estimation 42
3.4 Density Control via Velocity Field Generation 51
3.5 Numerical Simulations & Experimental Validation 59

Chapter 4: Collision Avoidance Analysis 71
4.1 Background 72
4.2 Problem Formulation 75
4.3 Analysis for 1-Dimensional Case 81

4.4 Conservative Generalization to Higher Dimensions 90

Chapter 5: Adaptive Density Estimation 96
5.1 Background 97
5.2 Problem Formulation in the Continuum 100
5.3 Finite-Agent Realization 107
5.4 Individual & Adaptive Estimation 111
5.5 Numerical Simulations 125

Chapter 6: Conclusion and Future Work 132
6.1 Concluding Remarks 132
6.2 Further Directions 133

Bibliography 135

Appendix A: Probabilistic Guidance Algorithm (PGA) 146

Appendix B: Heat Equation based Control 149
B.1 Exponential Convergence 149
B.2 Finite-Agent Analysis 151

Appendix C: Density as Kernel Mollification of Empirical Measures 155

Appendix D: Preliminaries on Functional Analysis & Semigroups 157
D.1 Functional Spaces and Operators 157
D.2 Semigroups 159

LIST OF FIGURES

Figure Number Page

1.1 A hierarchical swarm density coordination and control architecture: High-level Coordination provides motion commands(coordination inputs) for the vehicle controller and low-level control generates motion plans (control inputs) to follow these commands. 10

2.1 Illustration of the partitioned configuration space $\mathcal{R}$ with a continuous boundary $\partial\mathcal{R}$ into disjoint subregions R_i. 17

2.2 Interplay between high-level swarm coordination and low-level vehicle control 21

2.3 Partitioning for the Space Mirror Example and the Configuration of the Partitions with indices. All the density configurations and density constraints are deliberately chosen symmetric, hence only the green colored partitions are presented in the following figures. 32

2.4 ℓ_2 norm for density error in Markov steps with diagonal elements of the 19×19 Markov matrix and its spectrum for (i) unconstrained, (ii) density rate constrained and (iii) both density upper bound & rate constrained cases . . 32

2.5 Density evolution in Markov steps for (i) unconstrained, (ii) density rate constrained and (iii) both density upper bound & rate constrained cases . . 33

2.6 Snapshots from the ensemble of temporal agent locations and fractions of time spent in each subregion for single agent case 35

2.7 Time ensemble of randomized motion of single agent which generates ACL. For full video: `https://www.youtube.com/watch?v=Ekpqa39LBd0` 36

3.1 The block diagram of the local density feedback control architecture for swarm of agents to track desired density commands. 42

3.2 Different Estimation Methods: Solely counting Eulerian approach (left), solely counting Lagrangian approach (middle), weighted counting Lagrangian approach (right). 43

3.3 An Estimation Example: Gaussian kernel function with its gradient (left), and the estimation performed at agent locations for $N = 500$ with the 2-dimensional Gaussian kernel function where h = 75 (right). 47

3.4 Smoothing for the Markov chain Probability Mass Function: Smoothing kernels are applied to the desired probability mass function obtained from Markov chain based density guidance (left), and continuously differentiable desired density profiles (right) are attained to be used in density feedback control. . 57

3.5 The swarm density distribution in time towards desired density. 60

3.6 The Mean Squared Error in time. 61

3.7 The evolution of the swarm through time illustrated in both LVLH frame and Scaled-Rotated coordinates. 66

3.8 Control signals in time with maximum magnitude over the swarm(red), mean(black), and min(blue). 67

3.9 Raw Desired Density Commands read from images of letters *A, C, L* and the smooth counterparts for the raw input commands 68

3.10 Deterministic Pattern Generation with 10 agents to form letter A, C and L respectively. For full video: `https://youtu.be/_Dxf_wx1l84` 69

3.11 Kernel Density Estimations using Gaussian Kernels during the *A, C, L* sequence over the domain that 10 Crazyflie quadrotors perform their maneuvers. 70

4.1 Parametrization of all 1-dimensional swarm configurations with respect to $N = 2M$, $P \in \{-M+1, -M+2, \ldots, M-1\}$, and $\epsilon_k \geq 0\ \forall k$. 81

4.2 Collision radius R_c with respect to interpolation length h for $L \in \{0.0001, 0.01, 1.0, 10.0\}$ and $N \in \{10, 30, 100\}$ cases. 88

4.3 *Top*: Simulations with different h parameters. For each $h \in \{1, 2, 3, 4, 6, 8, 12\}$ 50 simulations with different initial conditions were run, and minimum observed inter-agent distance was recorded and compared with what (4.22) predicts. *Bottom*: Output of one of the simulations for $h = 8$ case. Blue dots are the initial, red dots are the steady-state agent locations of (4.11) with density values calculated at agent locations, and dashed line is $\rho_D(x)$. 89

4.4 Exact (4.22) versus conservative (4.24) bounds on collision radius R_c with respect to interpolation length h for different number of agents N and desired density gradient Lipschitz constants L_{ρ_D} 93

4.5 Comparison between upper bounds of collision avoidance radius R_c from (4.25) and (4.26) with respect to interpolation length h for different number of dimensions d, number of agents N and desired density gradient Lipschitz constants L_{ρ_D} 95

5.1 Desired Density Profile 125

5.2 Evolution of the decentralized adaptive-bandwidth density controller for $N = 16$, 64, and 256 agents over a 100×100 domain. Each row corresponds to a different swarm size and each column corresponds to a snapshot in time. Colored regions denote the swarm density $\rho(\mathbf{x}, \mathcal{X}, \bar{h})$ and black markers indicate agent locations. 126
5.3 Mean bandwidth evolution for the decentralized adaptive density controller with $N = 16$, 64, and 256 agents. Solid curves show the mean bandwidth $\bar{h}(t)$, while shaded envelopes indicate the range $\min_i h_i(t) \leq h_i(t) \leq \max_i h_i(t)$. 127
5.4 Comparison of steady-state density representations obtained using fixed and adaptive bandwidths for a swarm of $N = 32$ agents. The fixed-bandwidth controllers employ constant kernel scales $h = 6$, $h = 7.5$, and $h = 9$, while the adaptive controller evolves the bandwidth according to (5.32) and converges to $\bar{h} \approx 6.23$. 129
5.5 Performance comparison between fixed and adaptive bandwidth density controllers for $N = 32$ agents. The upper panel shows the density-error functional $\Phi(\mathcal{X}(t), \bar{h}(t))$ for three fixed bandwidth values and the proposed adaptive bandwidth strategy. The lower panel shows the corresponding adaptive bandwidth evolution, where the solid curve denotes the mean bandwidth $\bar{h}(t)$ and the shaded region indicates the range $\min_i h_i(t) \leq h_i(t) \leq \max_i h_i(t)$. . . 130

LIST OF TABLES

Table Number Page

3.1 Examples of Kernel Functions with Fourier Transform and Roughness Parameters 48

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor and dissertation committee for their guidance, support, and insightful feedback throughout this work and my doctoral studies. I also thank my colleagues and collaborators for their valuable discussions and contributions. Finally, I am grateful to my family and friends for their continued encouragement and support.

DEDICATION

To those who plant trees whose shade they know they shall never sit in.

Chapter 1

INTRODUCTION

The concept of deploying a sheer number of autonomous vehicles has emerged as a new paradigm over the last several decades. Control of these autonomous multi-vehicle systems has attracted a significant research interest due to the new capabilities and robustness that it provides in numerous engineering applications such as swarms of micro vehicles for surveillance, large distributed RF/microwave arrays, ground or aerial vehicle swarms for search and rescue, and mobile aerial, naval, land or space-based sensor networks. Employing many low-cost agents rather than more capable yet expensive few vehicles has self-evident benefits on redundancy, reconfigurability and parallelism. For instance, losing a highly capable single vehicle due to a failure or adversarial action can terminate a mission, whereas the loss of several vehicles in a swarm can be compensated without jeopardizing the mission. Swarms can dynamically form different patterns/configurations based on evolving mission requirements, and by balancing computational resources over a number of vehicles, different tasks can be operated simultaneously. Furthermore, the ability to recover from loss of several vehicles opens up a room for disposable vehicle concept while facilitating various new strategic mission scenarios.

Despite its benefits, guiding and controlling a swarm of vehicles to achieve desired formations and coordinated maneuvers introduces significant challenges. First, the guidance and control architecture is required to handle large ensemble of vehicles, i.e., algorithmic scalability is essential to enable control of swarms. Therefore, decentralization is crucial as each agent/vehicle comes with limited resources in computation, communication, sensing, and actuation. Second, robustness is critical: swarms must adapt to dynamic environmental changes (e.g., external disturbances) and internal modifications (e.g., agent loss or addition) relying only locally available information. Finally, ensuring collision avoidance and mitigating conflicts within performance envelope of agents for large-scale swarms is a crucial yet a formidable requirement for many swarm applications. Hence, the lack of scalable, robust and readily available coordination algorithms continues to be one of the key bottlenecks of swarm technology.

1.1 Research Motivation and Objectives

As autonomy advances further and becomes more prevalent, many scientific missions increasingly rely on more sophisticated and capable systems. Ranging from robotic arms to probes and rovers, robots continue to be an essential part of such missions and are expected to play a significant role in future endeavors. Swarm robotics is poised to play an important role in these future settings, offering capabilities that monolithic systems intrinsically lack. In fact, swarm intelligence [16] has emerged as an interdisciplinary field, with applications spanning engineering [25, 123, 15, 57], physics [127], biology [64], sociology [70], and more. Recent proposals for scientific missions leveraging swarm technology include NASA's "Orbiting Rainbows" concept [14] for space-borne exoplanet imaging telescope, and swarm robotics-based designs submitted by four finalist teams [1, 2, 3, 5] competing for "Shell Ocean Discovery XPRIZE" [4]. The former mission concept seeks to address the challenges and costs associated with launching and constructing large, complex space telescope mirrors, while the latter aspires to new deep-sea technologies for scalable autonomous high-resolution ocean exploration. Additionally, an interesting space exploration mission concept is proposed for micro-gravity environments such as surface exploration of small bodies, comets and asteroids [30, 106]. Utilizing multi-agent system of internally actuated spacecraft/rover hybrids that makes "hopping" maneuvers on the surface, the project envisions a mission architecture that is technically feasible and cost efficient for scientific explorations in low gravity environments.

The growing interest in swarm technology comes with distinctive engineering challenges in various aspects such as providing sufficient sensing, perception & communication systems, robust and distributed coordination & control algorithms and integration of human operators and teammates [81]. The objective of this research is to develop an architecture and its necessary algorithmic components for coordinating and controlling swarms of tens to thousands of autonomous vehicles using the concept of density. Specifically, this research will address two main challenges: (i) specification and synthesis of desired target configuration of the swarm, (ii) autonomous control of each unit to achieve the target swarm configuration. The following criteria are considered to be *primitives for swarm guidance & control* and will be the point of interest for finding a systematic way to address aforementioned challenges:

- *Scalability & Decentralization*: Multi-agent systems with sheer number of agents mandate scalable coordination and control algorithms. Since the dimensions of the swarm state space (joint state space of all agents) increases with the number of agents, the computational complexity of the coordination problem suffers from the population size, rendering many centralized algorithms computationally intractable. Furthermore, concentration of the information from all the agents to a centralized unit may not even be possible for some applications due to limited sensing and communication capabilities of each agent. Most importantly, if the centralized unit fails to deliver computations, the whole mission is jeopardized as there is no notion of redundancy. Therefore, decentralization of computations and relying on local information is crucial for real world swarm applications.

- *Safety*: Safety of the swarm can be elaborated two-fold: (i) safety against hazardous or uncertainty induced factors from the environment, (ii) safety against the internal interactions between agents. Swarm may have to operate in a domain where certain regions are risky for the operation of agents or less favorable for exploration. Hence, the probability (or frequency) of visiting that region is desired to be low compared to the rest of the domain. Similarly, due to the characteristics of regions (lighting conditions, topological features, GPS coverage, limited communication etc.), uncertainties may emerge in sensing and perception, thus, it is aspired to have less agents working simultaneously in some regions to eliminate conflicts due to uncertainties. These high-level environmental concerns must be addressed along with safe agent interactions within the swarm via generation of collision/conflict free trajectories through vehicle level control policies. Therefore, it is imperative for vehicle level control algorithm to be aware and respectful of high-level specifications while performing local maneuvers.

- *Resource Optimality*: Resource management is a significant concern in many aerospace applications. Especially in cases where the lifespan of the mission is determined by the amount of fuel/power within the system, efficient use of the resources becomes mandatory. In a multi-agent system, achieving a certain goal can be accomplished in various different ways as the number of agents in the system provides more degree of freedom than a monolithic system. Leveraging this degree of freedom to obtain resource efficient decision and control mechanisms not only extends the mission lifespan but also reduces the cost of many applications.

- *Mathematical guarantees for desired system behavior*: Guidance and control systems are typically desired to have mathematical guarantees to ensure satisfactory performance. The mathematical results covers variety of notions, including, but not limited to, convergence of the algorithm, stability of the system, obeying vehicle constraints, optimal use of resources, collision avoidance, robustness to uncertainties etc. In case of swarm guidance and control, a new set of performance metrics emerges such as achieving a desired swarm behavior, being able to reconfigure with evolving specifications & conditions.

- *Consideration of agent constraints*: Each agent in a swarm is subjected to physical limitations. In other words, desired trajectory or motion commands that are required by the high-level coordination commands might be infeasible due to the constraints on actuation capabilities of each agent. Therefore, the coordination algorithm must account for these limitations when generating feasible control commands.

Addressing these criteria will guide the development of swarm control systems that are both effective and adaptable to a wide range of mission requirements. Taking these criteria as a guiding framework, this research aims to lay the groundwork for swarm control systems that not only meet the technical demands of scalability, safety, and resource efficiency but also enhance the practical deployment of swarm technologies across practical mission scenarios. With primary focus being space applications ranging from Low Earth Orbit (e.g., spreading high-speed Internet with swarm of satellites) to deep space and planetary explorations (e.g, dozens of small probes to explore asteroids, comets), the outcome of this research can also be beneficial for aerial, naval or land-based surveillance, search and rescue/recovery applications along with wild-land firefighting and agricultural operations.

1.2 State Of The Art

The concept of swarm intelligence stems from observations of natural systems, such as bird flocks, fish schools, and ant colonies, where complex behaviors emerge from simple local rules. Inspired by these natural phenomena, early computational models for distributed systems paved the way for swarm robotics and autonomous multi-agent systems [20]. These systems exemplify the transformative potential of biologically inspired strategies in addressing technological challenges, bridging the gap between natural intelligence and artificial systems. Early research in the 1980s and 1990s laid the foundation for swarm robotics by exploring self-organizing behaviors, drawing from fields such as artificial intelligence, control theory, and biology. A landmark in the development of the field was the implementation of reactive robots, exemplified by Rodney Brooks' subsumption architecture [27], which demonstrated how simple, behavior-based control could enable robots to perform tasks autonomously in dynamic environments. In [28], Brooks suggested that rather than relying on isolated cognitive models, one should focus on creating systems from a collection of simple machines that operate without shared representations or central control. These systems should function within complex environments and have their complexity gradually increased over time. While this approach may not necessarily produce the "optimal" form of intelligence in some sense, Brooks argued that it could offer a viable solution where other strategies have failed. With the advent of more powerful computational tools and advancements in communication technologies, swarm robotics has grown significantly in scope and application. The early 2000s saw the deployment of robotic swarms in experimental scenarios, such as the SWARM-BOTS project [47], which demonstrated cooperative object transport and path finding. As the field continued to evolve, research shifted toward scalability, robustness, and fault tolerance, enabling larger and more complex swarms [25]. In recent years, swarm robotics has been proposed for diverse domains, including search-and-rescue operations [115], environmental monitoring, and space exploration, demonstrating the potential of decentralized systems to tackle real-world challenges.

Building on the foundational work of early swarm robotics and the principles of biologically inspired strategies, the focus of swarm behavior in multi-agent systems has increasingly centered around replicating the emergent behaviors observed in natural systems. From these natural models, three key principlesflocking, formation control, and self-organizationhave emerged as the cornerstones for designing and controlling artificial swarms.

Flocking refers to the process by which agents align, attract, and separate to achieve cohesive group motion. Inspired by Reynolds' classic "boids" model [112], flocking is typically governed by three simple rules: (*i*) separation rule for avoiding collisions with nearby agents, (*ii*) alignment rule for steering towards the average heading of neighboring agents, and finally, (*iii*) cohesion rule for moving towards the average position of nearby agents. These rules enable swarms to maintain unity while navigating through dynamic environments, making flocking a key behavior in applications like drone swarms and satellite constellations. Flocking models have been extended to include external influences such as environmental variations [87, 99] and leader-driven dynamics [121], enhancing their utility in goal-oriented missions.

Formation control focuses on arranging agents into specific spatial patterns, such as lines, grids, or complex shapes, to accomplish mission objectives. This principle is critical in applications like surveillance, where agents must maximize coverage, or communication networks, where maintaining line-of-sight is essential. Techniques for formation control often rely on graph theory, where agents are modeled as nodes and their interactions as edges [93]. Consensus-based protocols [84] are widely used to achieve and maintain desired formations. Some techniques also have integrated formation control with intrinsic network connectivity constraints to adapt patterns dynamically based on mission requirements [78].

Self-organization enables swarms to exhibit robust adaptation to dynamic environments without centralized control. This principle leverages local interactions to produce emergent global behaviors, such as clustering, dispersion, or synchronized oscillations. Self-organization is particularly valuable in unstructured or uncertain environments, where predefined control strategies may fail. Biological models, such as ant foraging and termite nest building, have inspired self-organizing algorithms in artificial swarms. For instance, pheromone-based models are used in robotic swarms to coordinate tasks like search and rescue [60]. Information consensus algorithms [111] and game-theoretic approaches [117] further enhance self-organization by enabling agents to negotiate and adapt to changing conditions in real time.

Control of multi-agent systems requires decision-making strategies that enable individual agents with limited capabilities and resources to achieve collective objectives. Existing approaches to swarm modeling and control can be broadly categorized into two perspectives: (i) microscopic point of view considers each agent as a fundamental unit of the system and models the behavior of each agent along with interactions between the agents; (ii) macroscopic point of view considers a high-level abstraction of the system and describes the collective behavior of the whole swarm rather than modeling agents individually [67]. Numerous microscopic control approaches have been proposed, such as leader-follower solutions [45, 131], where the swarm characterizes its motion based on trajectories of leader agents, as well as algorithms that leverage the notion of consensus over networked dynamical systems [93, 33, 32, 59, 132, 100]. Other methods include artificial potential field approaches [31, 13, 61, 75], distributed adaptive strategies for obstacle avoidance [72], and Coulomb's law-based nonlinear control techniques using electrostatic forces for spacecraft formation control [122, 116]. Miscellaneous agent-level methods are compiled in [136]. While effective for coordination and collision avoidance, most microscopic approaches emphasize inter-agent distances, local interactions, or explicit waypoint assignments [96], which limits their scalability and suitability for high-level tasks such as autonomous pattern generation, spatial deployment, and complex collective behaviors in large populations.

Macroscopic models, by contrast, primarily focus on the collective behavior of the swarm and are particularly suited for large populations consisting of hundreds or thousands of agents, where individual interactions become computationally intractable to model. In this framework, the swarm is treated as a continuum and its evolution is governed by partial differential equations (PDEs), mean-field models, or probabilistic density dynamics. For example, macroscopic swarm dynamics have been modeled using the advection equation [56], reaction-diffusion equations for deployment and self-organization tasks [110, 76], and incompressible flow formulations to regulate collective motion and spatial distributions [108, 109]. Recent work has also explored high-dimensional continuification and mixed-reality environments to bridge discrete swarm systems and continuum control models [91]. A comprehensive overview of such mean-field and continuum-based approaches is provided in [49]. Density-based control has attracted increasing attention due to several advantages it offers. First, the notion of density admits both probabilistic and deterministic interpretations, enabling applications ranging from stochastic coverage and surveillance (in time domain for a low number of agents) to deterministic pattern formation (in spatial domain for a large number of agents). Second, density-based formulations allow mission specifications to be expressed directly at the swarm level, facilitating the synthesis of complex collective behaviors while abstracting away individual agent interactions. Furthermore, density regulation can implicitly encode agent-level constraints such as collision avoidance when appropriately formulated. Early examples of density control using repulsive and attractive interactions for group motion and segregation are presented in [133]. More recent work has explored density stabilization on manifolds for nonholonomic agents [50], pseudo-localization based density control for self-organizing swarms without global positioning [82, 83], and probabilistic density evolution for stochastic coverage tasks [48]. PDE-constrained optimization frameworks have been proposed for density regulation of large-scale robotic swarms [119], along with robust and predictive extensions based on mean-field and model predictive control techniques to explicitly address uncertainty and disturbances in swarm dynamics [120, 36].

Optimal transport (OT) theory has emerged as a powerful tool for density-based swarm guidance, providing a principled framework for steering swarm distributions between desired configurations. Probabilistic swarm guidance using OT has been investigated in [12], with extensions to swarm-to-swarm engagement scenarios [124] and decentralized state-dependent Markov chain synthesis for swarm guidance [125]. Gradient-flow formulations of density propagation in stochastic systems further strengthen the connection between OT and macroscopic swarm control [29], while discrete-time linear-quadratic regulation via optimal transport establishes links between classical control theory and distributional dynamics [38]. These approaches have also been extended to incorporate temporal logic specifications [46] and predictive pattern formation in collective motion [130].

1.3 Proposed Approach

As the number of agents increases, the complexity of the microscopic models grows, hence, establishing stability and convergence characteristics of the overall swarm in a verifiable manner becomes arduous. Therefore, for guidance and control of swarms with very large number of agents, macroscopic models are preferred over microscopic models. Sole utilization of macroscopic models, on the other hand, generally fails to capture inter-agent interactions and hinders strong guarantees for conflict/collision avoidance. As it is illustrated in Figure 1.1, the guidance and control architecture in this research presents a hierarchical utilization of both models to utilize desirable properties of each approach.

Density Guidance & Control Architecture

Decomposing the problem into hierarchical guidance and control architecture, and addressing each component separately while keeping the hierarchical framework coherent provides several advantages. First, the decomposition provides flexibility to use different mathematical models for each component, hence, desirable properties of different models can be leveraged simultaneously. Second, overall swarm specifications can be enforced and addressed in different levels of abstraction. For instance, while convergence to desired behavior or reconfigurability can be addressed at density guidance level, communication & sensing limitations and collision/conflict avoidance is enforced in the density control level. Finally verifiability of each component's characteristics is simplified, i.e., a tractable problem decomposition facilitates a tractable verification process.

Previous research [8, 9, 43, 44] has addressed the methodologies to design sequence of desired density commands with prescribed properties such as safety, motion, and flow constraints. The proposed method in this research specifically focuses on generating a vehicle level control signals, which uses the local density information as a feedback to achieve desired density commands provided by high-level coordination algorithm. The idea is that, desired

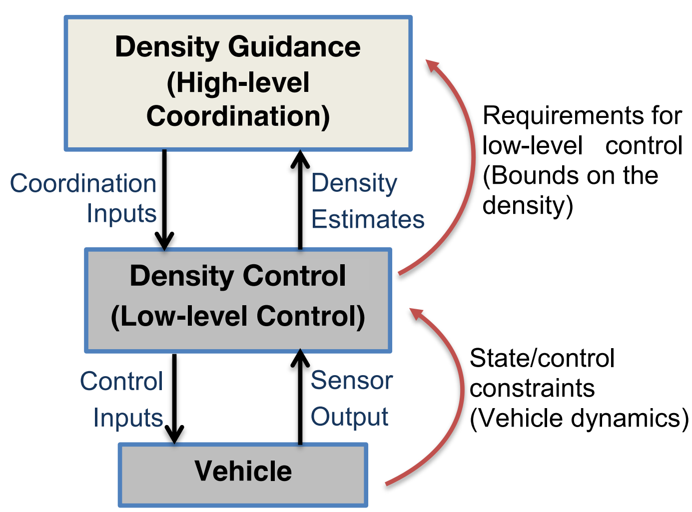

Figure 1.1: A hierarchical swarm density coordination and control architecture: High-level Coordination provides motion commands(coordination inputs) for the vehicle controller and low-level control generates motion plans (control inputs) to follow these commands.

macroscopic behavior, i.e., the sequence of density commands, can be produced by using the techniques presented in these earlier papers, and the density feedback law mentioned in [53] can be used to track them by generating microscopic signals, i.e, velocity fields, as a function of the desired density and the current local density estimate. Hence, this approach is made feasible due to a coherent combination of macroscopic and microscopic models. The high-level density commands are generated considering the collective behavior of the swarm, while local density information is obtained by interactions between agents. To this end, formulating the swarm control problem as a density control problem is the first key step of the proposed algorithm.

1.4 Thesis Outline

Building on the challenges and opportunities discussed in this chapter, this dissertation develops a hierarchical, density-based guidance and control framework that integrates high-level probabilistic density synthesis with decentralized feedback control. The proposed approach is designed to achieve scalability, safety, and robustness while providing provable stability and convergence guarantees in the large-population limit. To this end:

- *Chapter 2* introduces the density guidance problem and presents a probabilistic framework for high-level swarm coordination. Desired swarm behaviors are specified in terms of spatial density distributions, and a constrained Markov chain synthesis approach is developed to generate admissible density evolution sequences that encode motion, safety, and flow constraints.

- *Chapter 3* addresses the decentralized density control problem, focusing on the design of local feedback control laws that enable individual agents to track desired density distributions. Inspired by the heat equation, the proposed density-based control method is formulated using nonparametric local density estimation, and stability and convergence properties are established in the continuum limit.

- *Chapter 4* investigates collision avoidance properties of density-based control laws and presents a formal analysis of a modified density-based feedback strategy. Sufficient conditions are derived under which inter-agent collisions are avoided. Theoretical guarantees are established for a class of kernel functions, providing lower bounds on the achievable minimum inter-agent distance under the proposed control law.

- *Chapter 5* focuses on adaptive local density estimation for decentralized swarm control. Fundamental relationships between weak formulations of PDEs, the heat operator, and the parameters of Gaussian kernel based local density estimators are established. Dynamic average consensus based estimation schemes are developed to improve density reconstruction and promote well-distributed swarm configurations.

Finally, Chapter 6 concludes the dissertation with a summary of the main contributions and discusses potential directions for future research in density-based swarm guidance and control.

Chapter 2

DENSITY GUIDANCE PROBLEM

This chapter addresses the problem of guiding large-scale swarms by regulating their spatial density through probabilistic state transitions. Density guidance determines the desired evolution of the swarm density, while subsequent chapters develop feedback control laws that realize these density commands through agent motion. Rather than prescribing individual agent trajectories, the proposed approach models swarm evolution at the distribution level using Markov chains, enabling a scalable guidance framework that is robust to agent-level uncertainties and heterogeneity. By formulating swarm guidance as a density steering problem over a finite state space, Markov chain models provide a principled means to encode motion constraints, environmental structure, and safety requirements directly into the transition dynamics. This chapter presents a constrained Markov chain synthesis framework for density-based swarm guidance. It begins with the necessary background, followed by a formal problem formulation and the introduction of the Markov Chain Synthesis Problem with constraints. The proposed method is then illustrated through simulations and demonstrations that highlight convergence behavior, constraint satisfaction, and practical applicability, along with their temporal and spatial interpretations in swarm guidance scenarios.

2.1 Background

Throughout the chapter the following partial list of notation is used: $\mathbb{R}^d$ is the d dimensional real vector space; $\mathbb{N}$ is set of nonnegative integers, i.e., $\mathbb{N} = \{0, 1, 2, \ldots\}$; $\mathbb{N}^+$ is set of positive integers, i.e., $\mathbb{N}^+ = \{1, 2, \ldots\}$; $\mathbb{N}_n^+ = \{1, 2, \ldots n\}$; $\emptyset$ denotes the empty set; $\mathbf{0}$ is the zero vector/matrix of appropriate dimensions; v^T is the transpose of the vector v; $\|v\|$ is the 2-norm of the vector v; $\theta \in \mathbb{P}^n$ is said to be a *probability vector* if $\theta \geq \mathbf{0}$ and $\mathbf{1}^\mathsf{T}\theta = 1$;

I is the identity matrix; $\mathbf{1}$ and $\mathbf{0}$ are the one and zero matrices (or vectors) respectively with appropriate dimensions; e_i is a vector of appropriate dimension with its i^{th} entry $+1$ and rest zeros; $Q = Q^\mathsf{T} \succ (\succeq) 0$ implies that Q is a symmetric positive (semi-)definite matrix; $R > (\geq) H$ implies that $R[i,j] > (\geq) H[i,j]$ for all i, j; $R > (\geq)\mathbf{0}$ implies that R a positive (non-negative) matrix; For $P = P^\mathsf{T} \succ 0$, $\|v\|_P = \|P^{1/2}v\|$ where $P = P^{1/2}P^{1/2}$ is a factorization of P with $P^{1/2} = {P^{1/2}}^\mathsf{T} \succ 0$ (e.g., $P^{1/2} = U\Lambda^{1/2}U^\mathsf{T}$ where $P = U\Lambda U^\mathsf{T}$ is an eigenvector decomposition of P); $\otimes$ denotes the Kronecker product; $\odot$ represents the Hadamard (Schur) product.

2.1.1 Discrete-time Markov Chains

The following material reviews definitions and fundamental properties of finite-state Markov chains used throughout this chapter. In particular, concepts such as state space representations, transition matrices, stationary distributions, and irreducibility are introduced, along with several standard lemmas that support later statements on stability and constraint satisfaction.

Definition 1 *A Markov chain is a discrete-time stochastic process such that each random variable takes values in a discrete set and for all states the conditional probability distribution of the future states of the process (conditional on both past and present states) depends only on the present state (i.e. Markovian property).*

A Markov chain can be represented with three parameters $\mathcal{S}$, M and $x(0)$ [11]. Here, $\mathcal{S}$ is the finite set of states $\mathcal{S} = \{s_1, \ldots, s_n\}$, that is, there is a discrete state space with cardinality n and s_j is referred to as "j^{th} state"; M is the the state transition matrix, which will be referred to as *Markov matrix*, whose entries are defined as the transition probabilities; i.e.,

$$M[i,j] = \text{prob}\{s(t+1) = s_i | s(t) = s_j\}, \tag{2.1}$$

and $x(0)$ is a probability vector giving the initial state probability distribution, where $x(0)[i]$ gives the probability of the i^{th} initial state. A discrete-time system is considered where

$s(t) \in \mathcal{S}$ is the state of the agent at time $t \in \mathbb{N}$ and '$s(t) = s_i$' is the event that the state is the i^{th} state at time t. Then, the probability density distribution $x(t) \in \mathbb{P}^n$ is defined as

$$x[i](t) = \text{prob}\{s(t) = s_i\}, \tag{2.2}$$

where t is the discrete time index and $x[i](t)$ is the probability of an agent to be in i^{th} state. Since $M[i,j]$ gives the probability of transition from state j to state i, Markov matrix M determines the time evolution of the probability distribution as

$$x(t+1) = Mx(t), \quad t \in \mathbb{N}. \tag{2.3}$$

If the transitional probabilities does not depend on time k, the Markov chain is called *stationary* or *homogeneous*. The well-established conditions that a Markov matrix should satisfy can be given as follows:

$$M \geq 0, \quad \mathbf{1}^\mathsf{T} M = \mathbf{1}^\mathsf{T} \tag{2.4}$$

Here the first property states that each entry of the Markov matrix must be nonnegative as it corresponds to a probabilistic quantity. The second property means that M is a column stochastic matrix and ensures that the sum of probabilities of transitions from state j to all the states in $\mathcal{S}$ is 1. Such matrices can be characterized as follows.

Definition 2 *Matrix $M \in \mathbb{R}^{n\times n}$ is a* Markov matrix*, $M \in \mathbb{P}^{n\times n}$, if $M \geq 0$ and $\mathbf{1}^\mathsf{T} M = \mathbf{1}^\mathsf{T}$.*

Note that the probability vector $x(t)$ stays normalized as $\mathbf{1}^\mathsf{T} x(t) = 1$ for all $t \geq 0$. This follows from the fact that $\mathbf{1}^\mathsf{T} x(0) = 1$ and $\mathbf{1}^\mathsf{T} M = \mathbf{1}^\mathsf{T}$, which implies that $\mathbf{1}^\mathsf{T} M^t x(0) = \mathbf{1}^\mathsf{T} M^{(t-1)} x(0) = \ldots = \mathbf{1}^\mathsf{T} x(0) = 1$.

Definition 3 *Consider the Markov chain M, as given in (A.1). A distribution $v \in \mathbb{P}^n$ is called a* stationary distribution *of M if it satisfies the equation*

$$Mv = v. \tag{2.5}$$

Clearly if v is the stationary distribution of M, then $x(t) = v, \quad \forall t = t_o, t_o + 1, \ldots,$ if $x(t_o) = v, \; t_o \in \mathbb{N}^+$. which follows from (A.1) and (2.5) . A matrix $M \in \mathbb{P}^{n\times n}$ that satisfies the constraint (2.5) is a column stochastic matrix where v is the eigenvector corresponding to eigenvalue 1 [73, 55].

Definition 4 *[86] A Markov chain M is called irreducible if for any two states $i, j \in \mathbb{N}_n^+$ there exists an integer t such that $M^t(i, j) > 0$.*

This means that it is possible to get from any state to any other state using only transitions of positive probability.

Lemma 1 *[86] Let M be the transition matrix of an irreducible Markov chain. There exists a unique probability distribution v satisfying $Mv = v$.*

Lemma 2 *[7] Consider the Markov chain M, with a stationary distibution v. Then for any probability vector $x(0) \in \mathbb{P}^n$, it follows that $\lim_{t\to\infty} x(t) = v$ for the system (A.1) if and only if spectral radius of $(M - v\mathbf{1}^T)$ is less than 1, i.e., $\rho(M - v\mathbf{1}^T) < 1$.*

2.2 Problem Formulation

The goal of the density guidance problem is to generate a sequence of density profiles that converges to a final desired density. One can consider the time evolution of density as a solution of a partial differential equation. However, when constraints on transient density profiles are involved with a task to converge to a specific final desired density, the synthesis of the evolution process for continuous density profiles becomes significantly challenging. To this end, in order to ease the incorporation of density constraints, infinite-dimensional density guidance process is transformed into a finite-dimensional synthesis problem by partitioning the configuration space $\mathcal{R}$ into m disjoint subregions $R_i, \; i = 1, \ldots, m$, i.e., $\mathcal{R} = \bigcup_{i=1}^{m} R_i$, and $R_i \cap R_j = \emptyset$ for $i \neq j$. As it can be seen from Figure 2.1, this decomposition represents the discretization of the continuous state-space, where each subregion represents a discrete state in the configuration space. Similarly, a density distribution $f_{\mathcal{R}}(\mathbf{x})$ can be stated as a

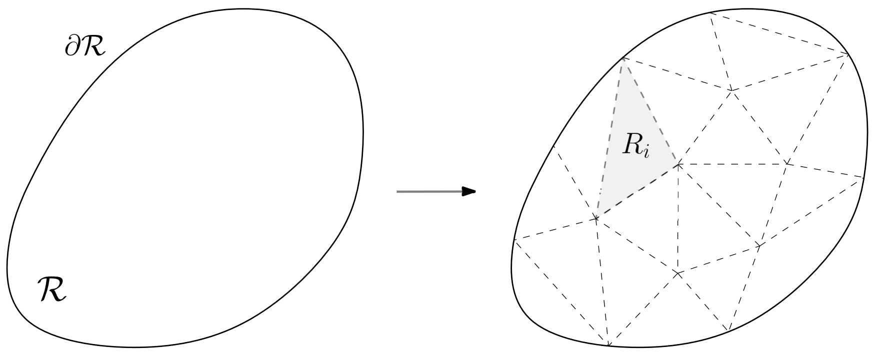

Figure 2.1: Illustration of the partitioned configuration space $\mathcal{R}$ with a continuous boundary $\partial\mathcal{R}$ into disjoint subregions R_i.

probability mass function $\theta \in \mathbb{P}^m$ over the partitioned space $\mathcal{R}$ as follows:

$$\theta[i] = \int_{R_i} f_{\mathcal{R}}(\mathbf{x})\mathrm{d}\mathbf{x}, \qquad \forall i \in \{1, \ldots, m\},$$

Now, consider a swarm of N agents. Let n^{th} agent state be $\mathbf{r}_n(k) \in \mathcal{R}$ at time index $k \in \mathbb{N}$, then we can define $\theta_n(k) \in \mathbb{P}^m$ such that the i^{th} element $\theta_n[i](k)$ is the probability of the event that $\mathbf{r}_n(k)$ is in subregion R_i at time index k, i.e.,

$$\theta_n[i](k) := \operatorname{prob}\big(\mathbf{r}_n(k) \in R_i\big), \quad i=1,...,m, \quad n=1,...,N, \tag{2.6}$$

where agents are assumed to be acting independently i.e., the probabilities of these N events are statistically independent. The discretization (2.6) facilitates a finite-dimensional problem formulation where constraints can be imposed directly on the vector $\theta_n(k)$.

Remark 1 *The density guidance algorithm feeds the sequence of probability mass functions $\theta_n(k)$ to density controller with a certain frequency. Therefore, the desired density commands are constructed as a piecewise continuous function in time as follows,*

$$\bar{\theta}_n(t) = \theta_n(k), \quad \forall t \in \big((k-1)\Delta t,\ k\Delta t\big], \quad k \in \mathbb{N}^+$$

where Δt is the duration between two consecutive desired density command updates.

2.2.1 Problem Definition

Let $\theta_n(0) \in \mathbb{P}^m$ be the initial probability mass distributions of agents in a swarm, given a specified steady-state swarm distribution $\nu \in \mathbb{P}^m$, the *density guidance problem* is defined as finding a sequence of $\theta_n(k)$ for all agents such that the following constraints are satisfied for all $k \in \mathbb{N}$:

$$\text{prob}\{\mathbf{r}_n(k+1) \in R_j \mid \mathbf{r}_n(k) \in R_i\} = 0 \quad \text{if} \quad A_a[i,j] = 0, \qquad \text{(Reachability)} \qquad (2.7)$$

$$L\theta_n(k) \leq p, \quad \forall \theta_n(k) \leq q, \qquad \text{(Density constraints)} \quad (2.8)$$

$$\lim_{k\to\infty} \theta_n(k) = v, \quad \forall \theta_n(0) \in \mathbb{P}^m, \qquad \text{(Ergodicity)} \qquad (2.9)$$

where A_a is the adjacency matrix defining the allowable transitions among subregions between two consecutive time indices, i.e., $A_a[i,j] = 1$ if the transition from subregion i to j is allowable, and is zero otherwise. L, q, and p are given matrices and vectors specifying density constraints. *Reachability constraint* prohibits undesirable or infeasible (within the maneuver envelope of each agent) transitions between subregions, whereas *density constraints* (also known as safety constraints) bound a prescribed linear functions of the sequence of desired density commands. Finally, *ergodicity constraint* ensures asymptotic convergence to prescribed steady-state final distribution. The density constraints are formulated in a general linear form and captures several types of constraints with different selection of the parameters. For example, they can be used to enforce an upper bound on the density or to bound the rate of change of density for each subregion.

2.2.2 Markov Chain Model

The main idea of the density guidance algorithm is to formulate the evolution of each agent's probability mass function $\theta_n(k)$ as a column stochastic matrix [17, 73], $M_n(k) \in \mathbb{R}^{m\times m}$ such that,

$$M_n[j,i](k) := \text{prob}\big\{\mathbf{r}_n(k+1) \in R_j \;\big|\; \mathbf{r}_n(k) \in R_i\big\} \quad \forall i, j = 1, \ldots, m, \quad \forall k \in \mathbb{N} \tag{2.10}$$

i.e., the n^{th} agent's state in subregion R_i transitions to subregion R_j between two consecutive time indices with probability $M_n[j,i](k)$. Note that $M_n(k) \geq \mathbf{0}$ and $\mathbf{1}^\mathsf{T} M_n(k) = \mathbf{1}^\mathsf{T}$, i.e., $M_n(k)$ is a column stochastic matrix, which we will also refer to as a *Markov matrix.* Consequently, for $k = 0, 1, \ldots$

$$\theta_n(k+1) = M_n(k)\theta_n(k), \quad n = 1, ..., N, \quad \text{with} \quad \theta_n(0) \in \mathbb{P}^m. \tag{2.11}$$

Note that the probability vector $\theta_n(k) \in \mathbb{P}^m$ for all $k \geq 0$ [6]. The Equation (2.11) shapes the evolution of each agent's probability mass function $\big\{\theta_n(k)\big\}_{n=1}^N$, rather than generating individual agent state trajectories $\big\{\mathbf{r}_n(k)\big\}_{n=1}^N$ [10].

Assumption 1 *From this point on, it will be assumed that each agent has the same Markov matrix at any given time index and same initial density distribution, that is, $M_n(k) = M(k)$, and $\theta_n(0) = \theta(0) = \theta_0$ $\forall n$.*

When all agents act independently by using an identical Markov matrix, we have two mathematical interpretations for $\theta(k)$:

(i) $\theta(k)$ is the vector of expected ratios of the number of agents in each subregion,

(ii) the ensemble of agent states, $\big\{\mathbf{r}_n(k)\big\}_{n=1}^N$, has a distribution that approaches $\theta(k)$ with probability one as N goes to infinity.

So the swarm probability mass function evolves with,

$$\theta(k+1) = M(k)\,\theta(k), \quad \text{with} \quad \theta(0) = \theta_0. \tag{2.12}$$

See the Lemma 11 in Appendix A for detailed analysis of these two interpretations.

2.3 Markov Chain Synthesis Problem with Constraints

Using the Markov Chain model, the density guidance problem defined in Section 2.2.1 is formulated as a Markov chain synthesis problem in which all desired constraints given by (2.7), (2.8) and (2.9) are imposed on the Markov matrix. Reachability constraints are easy to capture by construction and can be enforced with a linear equality condition on Markov matrix. Imposing ergodicity constraints on Markov chains is an addressed problem [86], and equivalent Linear-Matrix Inequality (LMI) conditions on Markov matrix has been presented in recent studies [7, 23, 24]. The density constraints, on the other hand, is a challenging formulation problem. For density upper bound constraints, a set of equivalent conditions on Markov chains were presented in [11], which are useful analysis results to check whether a given Markov chain satisfies the density upper bound constraints or not. The research presented in [9, 41] extends on these results by providing a computationally tractable convex analysis method but also, more importantly, a *convex optimization based synthesis* method. Then, our previous work [44] generalized these density upper bound constraints to a linear form that captures these constraints along with additional useful constraints. Before presenting the convex programming formulation of the Markov chain synthesis problem, the formulation and convexification of the constraints will be given.

2.3.1 Formulation & Convexification of the Constraints

Reachability Constraints

Reachability constraints are imposed on matrix $M(k)$ to restrict allowable agent motion or to ensure the generation of physically realizable agent motion commands by limiting the set of subregions reachable from each subregion. For example, it may not be desirable or even physically possible for an agent in R_j to transition to some other subregion R_i in a single density command update. Such transition constraints are captured by the adjacency matrix A_a as defined in (2.7) and can be enforced to Markov chain synthesis problem by setting the associated transition element of $M(k)$ to zero, i.e., $M(k)[i, j] = 0$. Noting that

A_a is assumed to be time-invariant, these transition constraints are imposed on $M(k)$ by the following equality constraint,

$$(\mathbf{1}\mathbf{1}^\mathsf{T} - A_a^\mathsf{T}) \odot M(k) = \mathbf{0} \qquad \forall k \in \mathbb{N}. \tag{2.13}$$

The entry $A_a[i,j]$ of A_a is defined for the transition from R_i to R_j while the entry $M(k)[i,j]$ is defined for the transition from R_j to R_i, hence the transpose operator for A_a is used in (2.13). It is also important to note that, in order to synthesize an ergodic Markov chain that is valid for any initial distribution, there has to be a sequence of subregions that connects

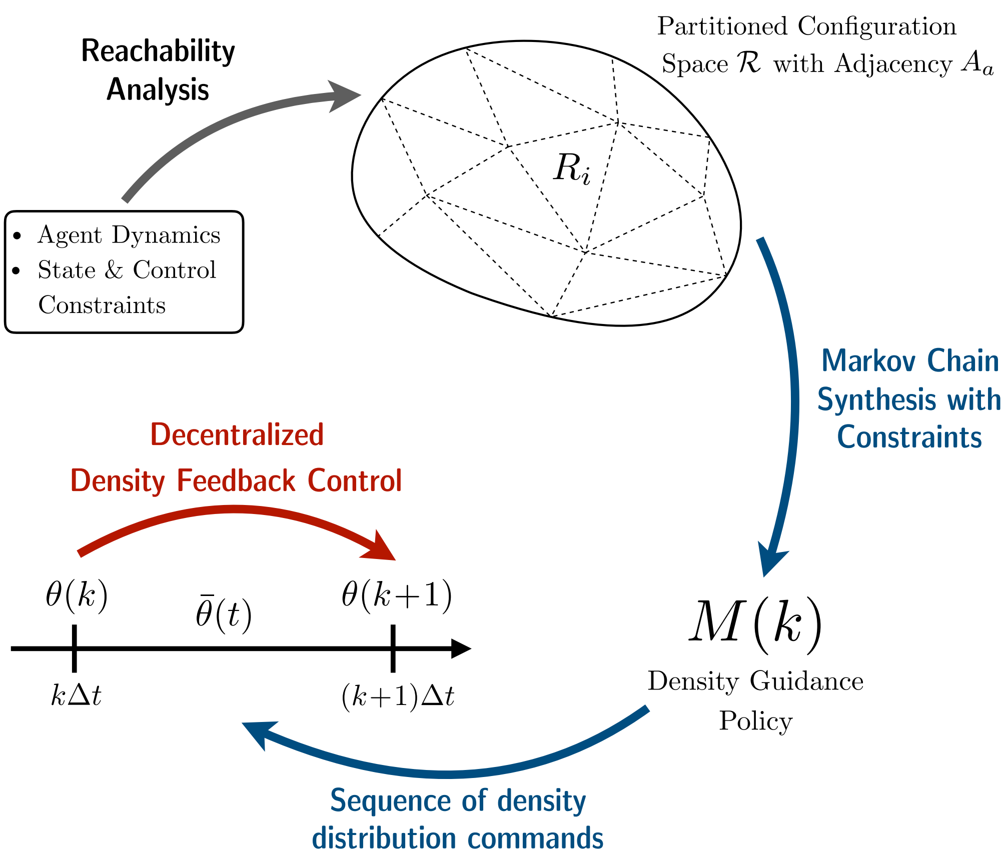

Figure 2.2: Interplay between high-level swarm coordination and low-level vehicle control

R_j to R_i for every pair (i, j). This can be ensured by selecting A_a as the edges of a strongly connected graph, i.e. for some integer k, A_a^k is a positive matrix. Since the Equation (2.13) is linear in $M(k)$, no convexification is required, i.e., it can be utilized in a convex programming formulation in its current form.

Density Constraints

The density constraints are captured in a generic form with the following set of inequalities

$$L(k)\,\theta(k) \leq q(k), \quad \forall\,\theta(k) \leq p(k), \quad k \in \mathbb{N}, \tag{2.14}$$

where, L is a linear function of M. Since $\theta(k)$ evolves with (2.12), the constraint (2.14) is a bilinear, non-convex constraint as both $M(k)$ and $\theta(k)$ are solution variables. The following theorem gives the convexification of these bilinear constraints by set of equivalent linear inequality constraints. As it is presented in [42], the theorem is proved by using the duality theory of convex optimization. The final form of the constraints presents an equivalent convex optimization formulation which enables a computationally tractable synthesis by using Interior Point Method (IPM) algorithms [98].

Theorem 1 *[9, 44, 42] Consider the Markov chain given by (2.12). Then,*

$$L(k)\,\theta(k) \leq q(k), \ \ \forall\,\theta(k) \leq p(k), \tag{2.15}$$

if and only if *there exist* $S(k) \in \mathbb{R}^{m\times m}$ *and* $y(k) \in \mathbb{R}^m$ *such that*

$$\begin{aligned} &S(k) \geq \mathbf{0}, \qquad \left[L(k) + S(k) + y(k)\mathbf{1}^T\right] \geq \mathbf{0}, \\ &y(k) + q(k) \geq \left[L(k) + S(k) + y(k)\mathbf{1}^T\right] p(k). \end{aligned} \tag{2.16}$$

Note that $p(k) = \mathbf{1}$ can be a choice in general, but other values of $p(k)$ will also be useful in capturing different types of density constraints. The bounds are generally taken as time-invariant but the proof given in [44, 42] is constructed for the time-varying case to provide most general linear result. It is also critical to note that the conditions given in (2.16) define linear inequality constraints on the solution variables, $S(k)$, $y(k)$, $M(k)$, if $L(k)$ is a linear function of $M(k)$, which is the main utility of the theorem. The density constraints given by (2.14) captures a variety of temporal and spatial density constraints with different selection of $L(k)$, $q(k)$ and $p(k)$.

1. *Density Upper Bound Constraint* [11, 9, 40]: These set of constraints ensure that the density vector components stay below the components of a prescribed values and they can be formulated with $L(k) = M(k)$, $q(k) = d(k{+}1)$, and $p(k) = d(k)$, which yields,

$$\theta(k) \leq d(k), \quad k = \mathbb{N}^+ \tag{2.17}$$

where $\mathbf{0} < d(k) \leq \mathbf{1} \;\; \forall k \in \mathbb{N}^+$ is a vector defining the prescribed density upper bounds for each subregion R_i and it is assumed that $\theta(0) \leq d(0)$. This constraint is also known as *safety constraint* in Markov chain literature [11] and it can be used to bound the expected number of agents (by using the expectation interpretation of the probabilistic density as explained by Lemma 11 in Appendix A) in each subregion over time. Invoking Theorem 1 with proper selections for $L(k)$, $q(k)$ and $p(k)$ yields the following corollary that states the equivalent linear matrix inequalities for the density upper bound constraints.

Corollary 1 *Consider the Markov chain given by (2.12). For every $\theta(0) \leq d(0)$, $\theta(k) \leq d(k)$, $\forall k \in \mathbb{N}^+$,* if and only if *there exists $S(k) \in \mathbb{R}^{m\times m}$ and $y(k) \in \mathbb{R}^m$ such that, for $k = 0, 1, \ldots$*

$$\begin{aligned} S(k) \geq \mathbf{0}, \qquad & \left[M(k) + S(k) + y(k)\mathbf{1}^T\right] \geq \mathbf{0}, \\ y(k) + d(k+1) \geq & \left[M(k) + S(k) + y(k)\mathbf{1}^T\right] d(k). \end{aligned} \tag{2.18}$$

2. *Density Flow Constraint*: Since the synthesized density commands are defined at discrete time instances, the density upper bound constraints given by (2.17) can be violated during

transitions between consecutive density commands $k \to k+1$. Although the density feedback controller is ultimately tasked to avoid collisions/conflicts due to high density of agents over a subregion, the density guidance algorithm must enable/facilitate the low-level controllers to perform the prescribed motion by producing commands that are consistent with density upper bound constraints. This objective can be achieved via an additional constraint on the flow through the subregions ensuring that the expected number of agents in each subregion is bounded by a prescribed quantity for all time - for a given upper bound on the total number of agents. Let $\bar{\theta}[i](k)$ be the maximum possible density for R_i during the time interval $\mathbf{I}(k)$ in between the temporal stages k and $k+1$, and $T_r \in \mathbb{R}^{m\times m}$ be a non-negative *trajectory matrix* for each subregion that describes the feasible paths containing subregion R_r, $r \in \{1,\ldots,m\}$: For $i,j \in \{1,\ldots,m\}$,

$$T_r[i,j] = \begin{cases} p_r[i,j] & \text{if path } j \to i \text{ passes through } R_r \\ 0 & \text{otherwise} \end{cases}, \tag{2.19}$$

where $p_r[i,j] \in [0,1]$ is the probability that the path from $j \to i$ passes through R_r. Note that $p_r[i,j] = 1$ when there is only a single feasible path between subregions R_i and R_j that must go through the R_r. Equipped with the trajectory matrix, the maximum density in R_i at any time $t \in \big((k-1)\Delta t,\ k\Delta t\big]$ is bounded by the sum of the probability that an agent choses to stay in R_i, the probability that an agent from another subregion transits to R_i, and the probability that an agent passes through R_i,

$$\bar{\theta}[i](t) \leq \underbrace{\theta[i](k)}_{\text{agents in } R_i \text{ at } k} + \underbrace{\sum_{j\neq i} M[i,j](k)\,\theta[j](k)}_{\text{agents moving into } R_i} + \underbrace{\mathbf{1}^{\intercal}\big(T_i \odot M(k)\big)\theta(k).}_{\text{agents passing through } R_i} \tag{2.20}$$

This inequality can be written in a matrix form for all subregions as follows,

$$\begin{aligned}\bar{\theta}(t) &\leq \theta(k) + M(k)\theta(k) - \mathrm{diag}\big(M(k)\big)\theta(k) + \sum_{i=1}^{m} \mathrm{e}_i \mathbf{1}^\mathsf{T}\big(T_i \odot M(k)\big)\theta(k) \\ &= \left(I + M(k) - \mathrm{diag}\big(M(k)\big) + \sum_{i=1}^{m} \mathrm{e}_i \mathbf{1}^\mathsf{T}\big(T_i \odot M(k)\big)\right)\theta(k).\end{aligned}$$

Using this relation, we can put a bound on maximum density on every subregion as $\bar{\theta}(t) \leq \alpha(k)$ for all $k \in \mathbb{N}$. With the assumption that (2.17) holds, we can write the following:

$$\left(I + M(k) - \mathrm{diag}\big(M(k)\big) + \sum_{i=1}^{m} \mathrm{e}_i \mathbf{1}^\mathsf{T}\big(T_i \odot M(k)\big)\right)\theta(k) \leq \alpha(k), \quad \forall\, \theta(k) \leq d(k), \quad k \in \mathbb{N} \tag{2.21}$$

The density flow constraint given by (2.21) can also be captured by the general density constraint form (2.14) via the following selections:

$$\begin{aligned} &L(k) = I + M(k) - \mathrm{diag}\big(M(k)\big) + \sum_{i=1}^{m} \mathrm{e}_i \mathbf{1}^\mathsf{T}\big(T_i \odot M(k)\big), \\ &q(k) = \alpha(k), \qquad p(k) = d(k). \end{aligned} \tag{2.22}$$

Upon the selections given by (2.22), we can have the following corollary for density flow constraints.

Corollary 2 *Consider the Markov chain given by (2.12). For* $\theta(k) \leq d(k)$, $k \in \mathbb{N}$, *the constraint (2.15) with (2.22) holds* if and only if *there exist* $S(k) \in \mathbb{R}^{m\times m}$ *and* $y(k) \in \mathbb{R}^m$, *such that, for* $k = 0, 1, \ldots$

$$\begin{aligned} S(k) \geq \mathbf{0}, \quad \Gamma(k) \geq \mathbf{0}, \\ y(k) + \alpha(k) \geq \Gamma(k)d(k), \end{aligned} \tag{2.23}$$

where $\Gamma(k) = I + M(k) - \mathrm{diag}\big(M(k)\big) + \sum_{i=1}^{m} \mathrm{e}_i \mathbf{1}^\mathsf{T}\big(T_i \odot M(k)\big) + S(k) + y(k)\mathbf{1}^\mathsf{T}$.

3. *Density Rate Constraint*: Density rate constraint bounds the change of density between two consecutive density profiles, i.e., $\theta(k)$ and $\theta(k+1)$ over subregions and is stated as follows,

$$-f \leq \theta(k+1) - \theta(k) \leq f, \quad \forall\, \theta(k) \leq d(k), \quad k \in \mathbb{N},$$

where $f \geq 0$ bounds the density rate. The general density constraints can capture density rate constraints with the following selection of parameters,

$$L(k) = \begin{bmatrix} M - I \\ I - M \end{bmatrix}, \quad q(k) = \begin{bmatrix} f \\ f \end{bmatrix}, \quad p(k) = d(k). \tag{2.24}$$

4. *Density Diffusivity Constraint*: This constraint limits the spatial gradients of the density with a prescribed vector, $w(k) \geq \mathbf{0}$,

$$-w(k) \leq D_1\, \theta(k) \leq w(k), \quad \forall\, \theta(k) \leq d, \quad k \in \mathbb{N}^+$$

where D_1 is a discrete approximation to the first order spatial derivative. The density diffusivity constraint ensures that the density evolves smoothly without steep jumps in the density between neighboring subregions. This inequality can be captured by the general form by letting

$$L(k) = \begin{bmatrix} D_1 \\ -D_1 \end{bmatrix}, \quad q(k) = \begin{bmatrix} w(k) \\ w(k) \end{bmatrix}, \quad p(k) = d(k). \tag{2.25}$$

Remark 2 *Similar results can be obtained for the density rate and density diffusivity constraints by using Theorem 1 and selections defined by (2.24) and (2.25).*

Convergence to Desired Density – Ergodicity Constraint

As it is stated in Section 2.2.1, the sequence of density commands are required to converge to a final desired density. Therefore the Markov chain $M(k)$ must be synthesized such that for any given initial distribution $\theta(0) \in \mathbb{P}^m$, the steady-state distribution $v \in \mathbb{P}^m$ of $M(k)$ is the final desired density, i.e.,

$$\lim_{k\to\infty} \theta(k) = v, \quad \text{and} \quad M(k)\, v = v.$$

Note that for the Markov chain $M(k)$ the condition $M(k)\, v = v$ means v is the eigenvector of $M(k)$ corresponding to its largest eigenvalue one [73, 55]. $M(k)\, v = v$ is a set of linear (hence convex) equalities, however this constraint alone is not sufficient for convergence to distribution v. The convergence constraint can be achieved by imposing a Lyapunov type condition as follows: For some $P = P^\mathsf{T} \succ \mathbf{0}$ and $\lambda \in [0, 1)$

$$\big\|\theta(k+1) - v\big\|_P \leq \lambda \big\|\theta(k) - v\big\|_P, \quad k = 0, 1, \dots. \tag{2.26}$$

which implies $\big\|\theta(k) - v\big\|_P \leq \lambda^k \big\|\theta(0) - v\big\|_P$, i.e., exponential convergence to v from any $\theta(0)$, therefore $\lim_{k\to\infty} \theta(k) = v$. In the special case when the Markov matrix $M(k)$ is constant with respect to time index k, Lemma 2 established a necessary and sufficient condition for asymptotic convergence on the design of matrix M, denoted as the *spectral radius condition* [7],

$$\rho(M - v\mathbf{1}^\mathsf{T}) < 1. \tag{2.27}$$

The synthesis of such constant Markov matrices also ensures the feasibility of online design of time-varying Markov matrices. From linear system theory [35, 79], if there exists a Lyapunov matrix $P = P^\mathsf{T} \succ 0$, the spectral radius condition (2.27) is equivalent to the following inequality for some $\lambda \in [0, 1)$:

$$\lambda^2 P - (M - v\mathbf{1}^\mathsf{T})^\mathsf{T} P (M - v\mathbf{1}^\mathsf{T}) \succeq \mathbf{0}. \tag{2.28}$$

This inequality is a bilinear matrix inequality with both M and P as solution variables. Inequality (2.28) can be shown to be equivalent to the following matrix inequality with some matrix G [39, 7]

$$\begin{bmatrix} \lambda^2 P & (M - v\mathbf{1}^\mathsf{T})^\mathsf{T} G^\mathsf{T} \\ G(M - v\mathbf{1}^\mathsf{T}) & G + G^\mathsf{T} - P \end{bmatrix} \succeq 0. \tag{2.29}$$

Here, the equivalence can be shown by applying Schur complement with respect to the block (2,2) and multiplying block (1,1) with $[I \;\; -(M - v\mathbf{1}^\mathsf{T})^\mathsf{T}]$ from the left and with its transpose from the right. In (2.29) G is a prescribed matrix, which is selected as $G = \mathrm{diag}(v)^{-1}$ in our implementations. To determine the λ, a line search can be conducted to find minimum feasible λ, hence maximum convergence rate. A more detailed discussion on the choice of G and λ can be found in [9]. It is important to note that prescribing the matrix G brings along a source of conservatism. Attempt to obtain matrix G from optimization can be achieved by solving Bilinear Matrix Inequalities (BMIs) with the expense of exponential computational complexity that grows with the problem size. Therefore it is only applicable to small size problems.

2.3.2 Markov Chain Synthesis with Semidefinite Programming (SDP)

All design constraints have been formulated as equivalent linear equality (2.13), linear inequality (2.16) and linear matrix inequality (LMI) (2.29) constraints. Therefore, an LMI optimization problem with the given constraints can be formulated to synthesize a Markov matrix. The optimization problem is a feasibility problem, however a minimization can be performed with a suitable cost function. For instance, observe that $M = I$ is a limiting case since it stops the density evolution based on (2.12), hence making $M \simeq I$ yields a sense of minimized overall density changes in each subregion, in other words, synthesizing M as close as possible to I minimizes the number of transitions between subregions. For this goal, the following cost can be minimized for minimal overall action [7]: $\mathbf{1}^\mathsf{T}\big(\mathbf{1} - \mathrm{diag}(M)\big)$. This cost function is particularly useful to be used with a randomized algorithm given in Appendix A where, even the convergence to desired density is achieved, the transitions still occur due to the randomized partition changing decisions. While perpetual partition changing decisions, therefore nonstop swarm motion, is not suitable for fuel efficient missions, it is a viable setup for surveillance applications. On the other hand, there are deterministic control algorithms such as in Chapter 3 which ensures that when the desired density is achieved the relative swarm motion ceases. For these type of control algorithms the guidance cost function of trace(M) is a fitting choice as it maximizes the number of partition changes and hence increases the convergence rate to the desired density. Both of these applications will be addressed both with numerical examples and laboratory demonstrations.

With a prescribed matrix G and a convergence rate $\lambda \in [0, 1)$, the following LMI optimization problem can be solved to synthesize a desirable Markov matrix when L, q, p in (2.15) are all constant quantities in each time indices:

Optimization Problem 1 *SDP Formulation for Markov Chain Synthesis*

Cost Function:	$\min\limits_{M,P,S,y} \quad \mathbf{1}^T\big(\mathbf{1} - \mathrm{diag}(M)\big) \qquad \text{s.t.}$
Markov Chain Properties:	$\mathbf{1}^T M = \mathbf{1}^T, \quad M \geq \mathbf{0}, \quad Mv = v$
Reachability Constraints:	$(\mathbf{1}\mathbf{1}^T - A_a^T) \odot M = \mathbf{0}$
Density Constraints:	$S \geq \mathbf{0}, \qquad [L + S + y\mathbf{1}^T] \geq \mathbf{0}$
	$y + q \geq [L + S + y\mathbf{1}^T]p$
Convergence Constraints:	$\begin{bmatrix} \lambda^2 P & (M - v\mathbf{1}^T)^T G^T \\ G(M - v\mathbf{1}^T) & G + G^T - P \end{bmatrix} \succeq 0$
	$P = P^T \succ 0 \qquad G = \mathrm{diag}(v)^{-1}$

Note that, in optimization problem above λ is considered to be known. This is a mild assumption since a line search can be performed for λ to find its smallest feasible value, i.e., the fastest convergence rate.

2.3.3 QP Synthesis of Markov Chain with Density Feedback

The Markov chain synthesis problem given in previous section is utilized in an open loop fashion, i.e., the Markov matrix M synthesized once and density commands are propagated with the same matrix throughout the swarm maneuver. When the measurements or estimates of the probabilistic density distribution are available, they can be utilized to update the Markov matrix, which leads to a faster convergence. The density of agent states are assumed to be given at this point, and for implementation purposes they can be estimated whether centrally or in some decentralized manner. Using this extra information for density, a receding horizon based synthesis of $M(k)$ can be formed for a given $\theta(k)$. Assuming that there exists a Markov matrix $\hat{M}$, a stochastic matrix satisfying the constraints of the Optimization Problem 1 with a corresponding positive definite matrix P, for a given $\theta(k)$, the following Quadratic Programming (QP) problem can be solved to find a new $M(k)$:

Optimization Problem 2 *QP Problem for Markov Chain Update with Density Feedback*

Updated Markov Chain:	$M(k) \;=\; \arg\min_{M} \;\; \\|M\theta(k) - v\\|_P \qquad \text{s.t.}$
Markov Chain Properties:	$\mathbf{1}^T M = \mathbf{1}^T, \quad M \geq \mathbf{0}, \quad Mv = v$
Reachability Constraints:	$(\mathbf{1}\mathbf{1}^T - A_a^T) \odot M = \mathbf{0}$
Density Constraints:	$L\,\theta(k) \leq q$

Optimization Problems 1-2 allow imposing the constraint $\theta(k) \leq d$, as in (2.17), together with $\hat{L}\,\theta(k) \leq \alpha$ with $\hat{L}$ as L in (2.22) with constant M, d, and α via

$$L = \begin{bmatrix} M \\ \hat{L} \end{bmatrix}, \quad p = d, \quad \text{and} \quad q = \begin{bmatrix} d \\ \alpha \end{bmatrix}.$$

The optimal solution of Problem 2 provides the *Markov matrix* $M(k)$ that leads to the closest density distribution to v, with respect to a norm defined by matrix P. Since, the optimization problem 2 is solved at each time index k, convergence and recursive feasibility of this feedback strategy is in question. Hence, the next lemma is provided, proof of which can be found in [44, 42].

Lemma 3 *Suppose that there exists a Markov matrix and $P = P^{\mathsf{T}} \succ \mathbf{0}$ satisfying the constraints of the optimization problem 1. Then the* Markov Matrix $M(k)$ *obtained by solving the QP given by Problem 2 for each $\theta(k)$ results in $\lim_{k\to\infty} \theta(k) = v$ for any $\theta(0) \leq p$ while satisfying the motion and general safety constraints with constant values of L, q, p in (2.15). Furthermore a feasible solution of the QP given by Problem 2 will exist for all time.*

According to Lemma 3, if there exists a Markov matrix $\hat{M}$, satisfying the constraints of the optimization Problem 1, then the convergence and the recursive feasibility of the proposed feedback method is guaranteed with the QP formulation given by Problem 2.

2.4 Numerical Simulations & Experimental Validation

2.4.1 Density Sequence Generation

The objective of this *space mirror* example is to illustrate the synthesis of density commands using the constrained Markov chain framework developed in the previous sections. In particular, the example demonstrates the simultaneous enforcement of reachability constraints, transient density constraints, and convergence to a prescribed steady-state density distribution. As it is illustrated in Figure 2.3 the operational domain is partitioned into 19 hexagonal bins. The adjacency structure is selected such that transitions are only permitted between neighboring partitions. The synthesized Markov matrix must respect the underlying motion limitations while steering the swarm from the concentrated initial configuration toward the desired deployment over the inner and outer layers. Density constraints are imposed throughout the maneuver to ensure that the generated density commands remain feasible for the lower-level density control algorithms responsible for realizing the desired swarm motion. To ease the illustration, symmetric safety constraints are applied and the results are presented only for a portion of the domain emphasized by green color in the Figure 2.3.

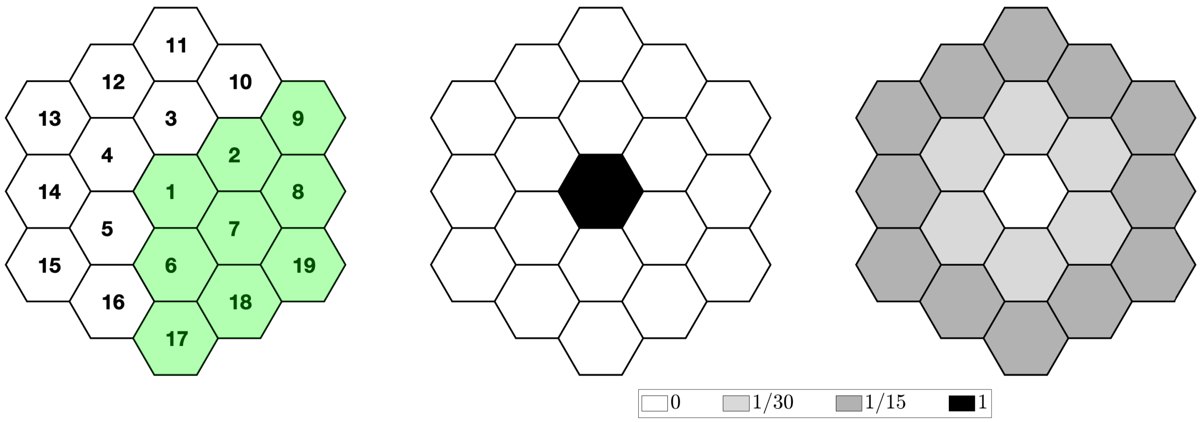

Figure 2.3: Partitioning for the Space Mirror Example and the Configuration of the Partitions with indices. All the density configurations and density constraints are deliberately chosen symmetric, hence only the green colored partitions are presented in the following figures.

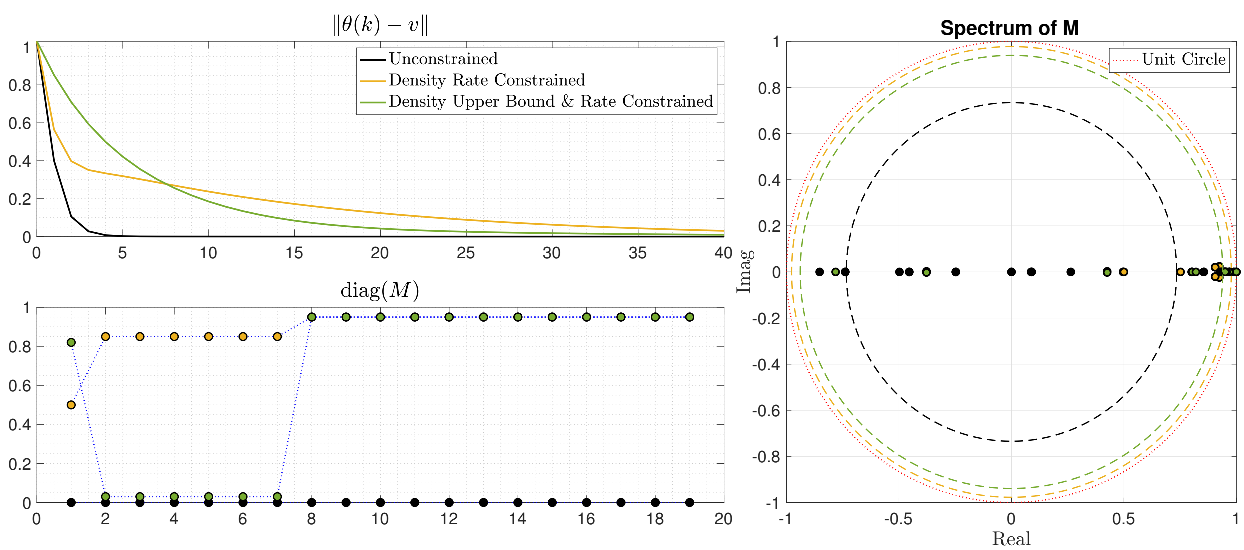

Figure 2.4: ℓ_2 norm for density error in Markov steps with diagonal elements of the 19×19 Markov matrix and its spectrum for (i) unconstrained, (ii) density rate constrained and (iii) both density upper bound & rate constrained cases

The initial and desired density configurations are also visually demonstrated in Figure 2.3. The spectral properties of the synthesized Markov matrices shown in Figure 2.4 provide additional insight into the effect of the imposed constraints. As additional constraints are introduced, the synthesized Markov matrices tend to exhibit eigenvalues that are closer to the unit circle, resulting in slower mixing and convergence rates. This behavior is reflected in the evolution of the density error norm. Nevertheless, all synthesized matrices retain the desired stationary distribution and satisfy the ergodicity conditions required for convergence. The mission is the deployment of arbitrary number of agents from the Partition 1 to other partitions labeled as *inner layer* and *outer layer*. The resulting density evolution is shown in Figures 2.5. The density propagates outward from Partition 1 through the admissible transitions while preserving the imposed constraints. The evolution exhibits the character-

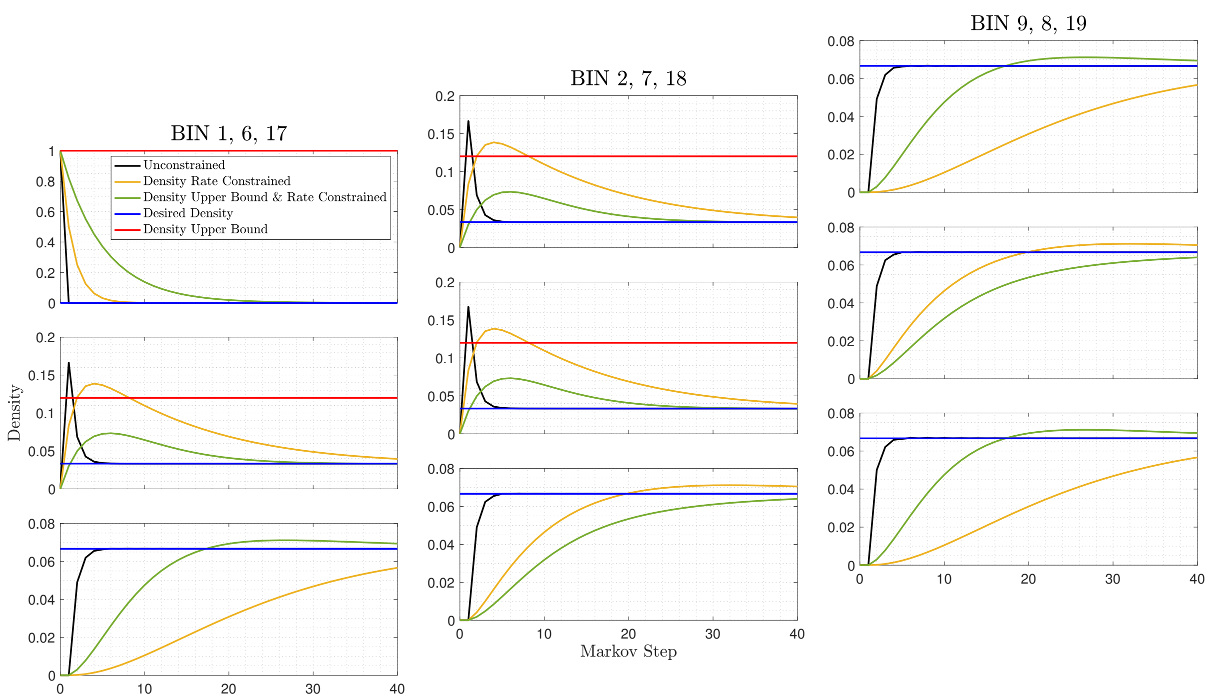

Figure 2.5: Density evolution in Markov steps for (i) unconstrained, (ii) density rate constrained and (iii) both density upper bound & rate constrained cases

istic behavior of a constrained diffusion process, where density can only propagate through admissible transitions and therefore spreads progressively from the source partition toward the target equilibrium configuration.

2.4.2 Stochastic Planning

This demonstration illustrates the use of the density guidance method for the single agent case via randomized motion driven by a Markov chain. In this example, the randomized motion of single agent solves a surveillance problem. In other words, an agent is instructed to cover the area that is discretized by 25 number of subregion as mentioned in Section 2.2 with equal probability. The motion of the agent is determined by the probabilistic guidance algorithm given in Appendix A. For this discretized domain, a Markov chain is synthesized whose stationary distribution produces the ACL pattern. Figure 2.6 illustrates the gradual emergence of the desired density distribution from the randomized motion of a single agent. An important distinction is that the desired density is not realized instantaneously. At any given time the agent occupies only a single partition. Instead, the density emerges through the long-term visitation statistics of the randomized trajectory. This behavior is consistent with the stationary-distribution properties of the synthesized Markov chain and demonstrates how density guidance can be interpreted through long-term occupancy statistics.

Rather than using multiple agents, the notion of density is satisfied on the expected duration that agent spends in each subregion, i.e., the agent traverses the domain while spending time in each subregion enforced by the desired density command. As it is further discussed in Appendix A via Lemma 11, the probabilistic interpretation of density in a subregion is the probability of finding an agent in that subregion, and this experiment given by Figure 2.7 demonstrates this interpretation. As time progresses, the fraction of time spent in each partition converges to the prescribed density distribution.

The experiment provides an intuitive visualization of the law-of-large-numbers interpretation discussed in Appendix A. In the multi-agent setting, the density represents the fraction of agents occupying each subregion at a given time. In contrast, for the single agent case, the

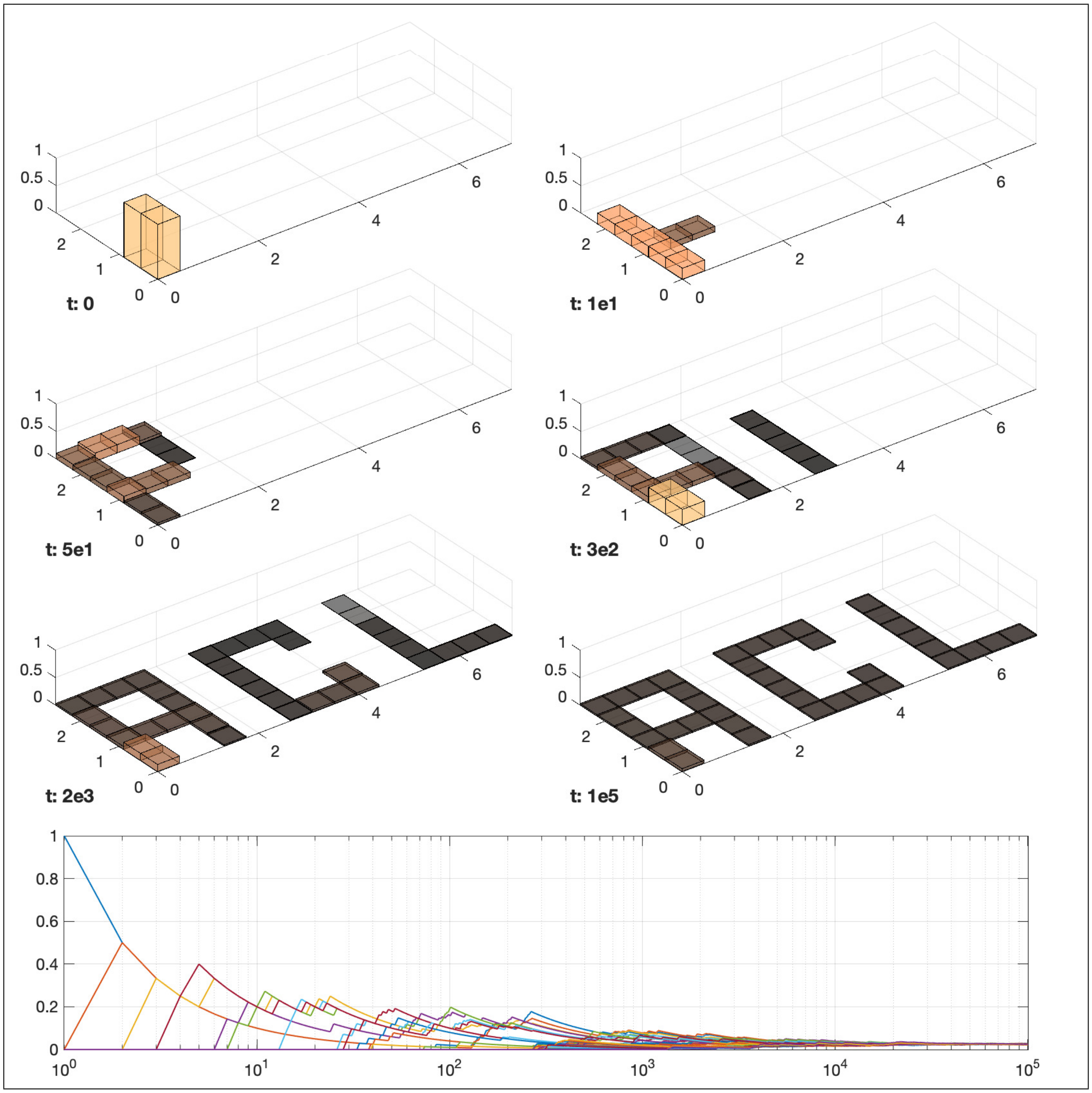

Figure 2.6: Snapshots from the ensemble of temporal agent locations and fractions of time spent in each subregion for single agent case

same density distribution emerges through temporal averaging of the agent trajectory. As the observation horizon increases, the empirical occupancy measure converges to the desired stationary distribution, producing the prescribed spatial pattern.

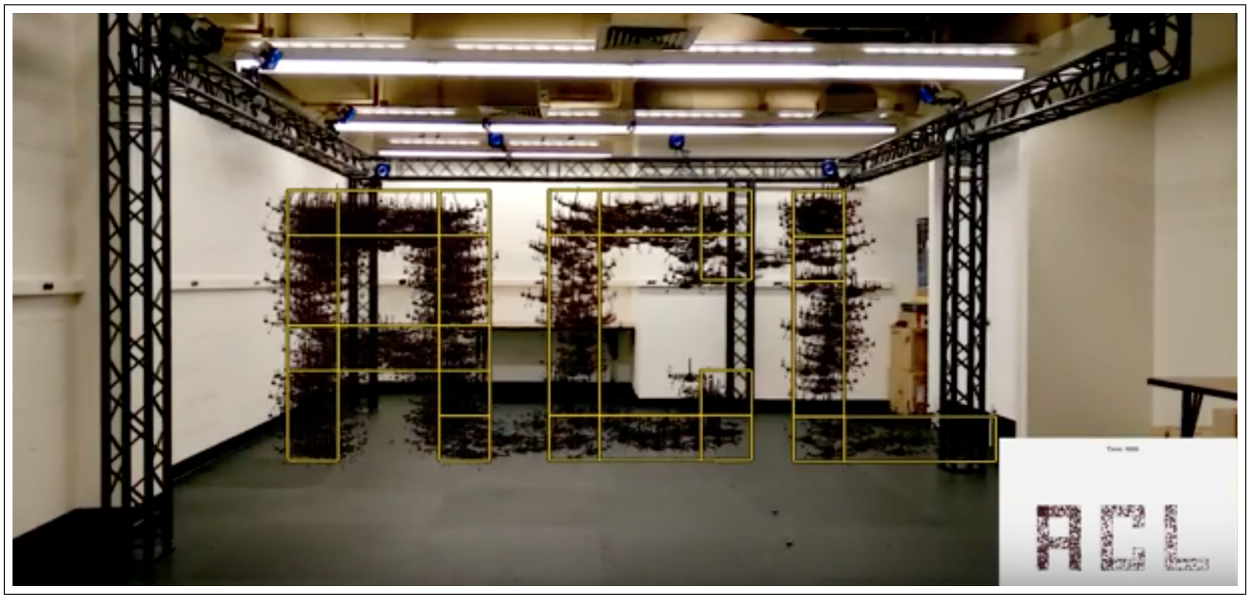

Figure 2.7: Time ensemble of randomized motion of single agent which generates ACL. For full video: `https://www.youtube.com/watch?v=Ekpqa39LBd0`

This example demonstrates that the proposed density guidance framework admits both ensemble and ergodic interpretations. In the multi-agent setting the density describes the spatial distribution of population of agents, whereas in the single agent setting the same density emerges through temporal averaging of a randomized trajectory. Therefore, the Markov chain synthesis framework is applicable not only to swarm deployment problems but also to stochastic planning and persistent surveillance missions.

Chapter 3

DENSITY FEEDBACK CONTROL

This chapter develops a density feedback control method for driving large-scale swarms through continuous-space interactions toward prescribed desired density profiles. In contrast to open-loop or distribution-level planning approaches, the proposed method regulates swarm behavior by closing the feedback loop between locally estimated density information and agent-level motion commands. The resulting closed-loop dynamics can be interpreted as a density evolution governed by velocity fields induced by local density errors. By constructing velocity fields directly from density errors, the framework enables decentralized and scalable control laws that remain responsive to environmental changes, disturbances, and modeling uncertainties. The chapter begins with the necessary background and problem formulation, followed by methods for local density estimation and density control via velocity field generation. Techniques for spatial smoothing of desired density profiles are then introduced to establish a connection with the previously developed density guidance methods. The effectiveness of the proposed approach is demonstrated through numerical simulations and experimental validation.

3.1 Background

This section provides a brief mathematical background regarding the necessary algorithmic components of the proposed density feedback control method. The following is a partial list of notation used: $\mathbb{N}$ is set of nonnegative integers, i.e., $\mathbb{N} = \{0, 1, 2, \ldots\}$; $\mathbb{N}^+$ is set of positive integers, i.e., $\mathbb{N}^+ = \{1, 2, \ldots\}$; $\mathbb{T}$, stands for range of time and unless otherwise is stated, it represents the nonnegative axis $[0, \infty)$; $\emptyset$ denotes the empty set; $\mathbf{0}$ is the zero vector/matrix of appropriate dimensions; v^T is the transpose of the vector v; $\|v\|$ is the 2-

norm of the vector v; $\mathcal{F}^\dagger$ is the adjoint of the operator $\mathcal{F}$; gradient operator is denoted as $\nabla = \left[\frac{\partial}{\partial x_1},\ \frac{\partial}{\partial x_2},\ \ldots,\ \frac{\partial}{\partial x_n}\right]^\mathsf{T}$, Laplacian is denoted as $\Delta(\cdot) = \nabla^\mathsf{T}\nabla(\cdot) = \sum_{i=1}^{d} \frac{\partial^2(\cdot)}{\partial x_i^2}$ and Hessian is denoted as $\mathcal{H}(\cdot) = \nabla^2(\cdot)$; $C^1(\mathcal{R}, \mathbb{R})$ represents the family of continuously differentiable scalar functions over $\mathcal{R}$; $\det(\cdot)$ stands for determinant operator; $\sup(\cdot)$ represent the supremum operator; $\mathbb{E}[\cdot]$ is the expectation operator; $\theta \in \mathbb{P}^n$ is said to be a *probability vector* if $\theta \geq \mathbf{0}$ and $\mathbf{1}^\mathsf{T}\theta = 1$;

3.1.1 Swarm Density Distribution

Consider a swarm of $N \in \mathbb{N}^+$ agents with point mass dynamics that are distributed over the configuration space $\mathcal{R} \subset \mathbb{R}^d$ with a continuous boundary $\partial\mathcal{R}$. The swarm density is defined as the fraction of agents per unit volume obtained in the limit $N \to \infty$, and it can be described at any point $\mathbf{x} \in \mathcal{R}$ at time $t \in \mathbb{R}^+$ as follows,

$$\rho(t, \mathbf{x}) \;=\; \lim_{\epsilon \to 0} \frac{1}{V_{\mathcal{B}_\epsilon(\mathbf{x})}} \left(\lim_{N \to \infty} \frac{n_{\mathcal{B}_\epsilon(\mathbf{x})}}{N} \right), \tag{3.1}$$

where $\mathcal{B}_\epsilon(\mathbf{x})$ is the ϵ-neighborhood of a point $\mathbf{x}$ defined as, $\mathcal{B}_\epsilon(\mathbf{x}) = \big\{\xi \in \mathbb{R}^d :\ \|\xi - \mathbf{x}\| < \epsilon\big\}$, in which $n_{\mathcal{B}_\epsilon(\mathbf{x})}$ is the total number of agents within $\mathcal{B}_\epsilon(\mathbf{x})$ and $V_{\mathcal{B}_\epsilon(\mathbf{x})}$ is the volume of $\mathcal{B}_\epsilon(\mathbf{x})$. Assuming the agent configurations are independent samples, the *law of large numbers* (Theorem 5.4.2 in [34]) implies that the remaining limit expression can be interpreted as the probability of having an agent in the ϵ-neighborhood of a point $\mathbf{x} \in \mathcal{R}$ at time $t \in \mathbb{R}^+$,

$$\lim_{N \to \infty} \frac{n_{\mathcal{B}_\epsilon(\mathbf{x})}}{N} \;=\; \mathrm{prob}\Big(\mathbf{r}(t) \in \mathcal{B}_\epsilon(\mathbf{x})\Big),$$

where $\mathbf{r}(t) \in \mathcal{R}$ is the configuration of an agent (can be position, velocity, heading angle, and so on). This probability can also be written in the following form,

$$\mathrm{prob}\Big(\mathbf{r}(t) \in \mathcal{B}_\epsilon(\mathbf{x})\Big) \;=\; \int_{\overline{\mathcal{B}}_\epsilon(\mathbf{x})} f_{\mathcal{R}}(t, \mathbf{x})\, \mathrm{d}\mathbf{x},$$

where, $\overline{\mathcal{B}}_\epsilon(\mathbf{x}) = \{\xi \in \mathbb{R}^d : \ \|\xi - \mathbf{x}\| \leq \epsilon\}$ is the closed neighborhood, $\mathrm{d}\mathbf{x} := \int dx_1 \int dx_2 \cdots \int dx_d$ and $f_\mathcal{R}(t, \mathbf{x})$ is the probability density function (PDF) that defines the distribution of the swarm over the space $\mathcal{R}$ and has positivity and unit integral properties:

$$f_\mathcal{R}(t, \mathbf{x}) \geq 0 \qquad \forall t \in \mathbb{R}^+, \ \forall \mathbf{x} \in \mathcal{R}, \tag{3.2a}$$

$$\int_\mathcal{R} f_\mathcal{R}(t, \mathbf{x}) \, \mathrm{d}\mathbf{x} = 1 \qquad \forall t \in \mathbb{R}^+. \tag{3.2b}$$

Here, the swarm with N agents at time t can be considered as a realization of the PDF $f_\mathcal{R}(t, \mathbf{x})$, i.e., N samples drawn from this PDF. In the context of swarm density control, $f_\mathcal{R}(t, \mathbf{x})$ will be referred to as the *swarm density distribution*. Since practical applications cannot accommodate the condition $N \to \infty$, one of the tasks in this research is to estimate $f_\mathcal{R}(t, \mathbf{x})$, or equivalently, approximate $\rho(t, \mathbf{x})$ efficiently.

Remark 3 *The definition of swarm density distribution is given with the assumption that the swarm is homogeneous i.e. all agents are identical. Although this assumption is adopted throughout this research, the framework can be generalized to swarms with heterogeneous agents using the notion of* weighted distributions *[103]. This approach adjusts $\rho(t, \mathbf{x})$ in (3.1) by putting an emphasis on each agent via assigned individual weights.*

3.1.2 Stochastic Density Evolution

The swarm density distribution introduced in the previous section can be interpreted as the probability density associated with the state of a representative agent evolving in configuration space. To establish a connection between individual agent motion and continuum swarm density evolution, consider the stochastic differential equation

$$\mathrm{d}\mathbf{X}_t = \mu(t, \mathbf{X}_t) \, \mathrm{d}t + \sigma(t, \mathbf{X}_t) \, \mathrm{d}\mathbf{W}_t, \tag{3.3}$$

where $\mathbf{X}_t \in \mathbb{R}^d$ denotes the agent state, $\mu(t, \mathbf{X}_t)$ is the drift vector field, $\sigma(t, \mathbf{X}_t)$ is the diffusion matrix, and $\mathbf{W}_t$ is a Wiener process. The probability density function associated with $\mathbf{X}_t$ is denoted by $f_{\mathcal{R}}(t, \mathbf{x})$. A classical result from stochastic process theory states that $f_{\mathcal{R}}(t, \mathbf{x})$ evolves according to the *Fokker-Planck* equation [90, 101],

$$\frac{\partial}{\partial t} f_{\mathcal{R}}(t, \mathbf{x}) = -\sum_{i=1}^{d} \frac{\partial}{\partial x_i}\big(\mu(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big) + \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d} \frac{\partial^2}{\partial x_i \partial x_j}\big(D_{ij}(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big). \tag{3.4}$$

where $D(t, \mathbf{x}) = \sigma(t, \mathbf{x})\sigma^\top(t, \mathbf{x})$ is the diffusion tensor.

This equation provides a continuum description of how a population of agents evolves under the combined effects of deterministic motion and stochastic disturbances, whereas (3.3) describes individual sample paths. The Fokker-Planck equation is introduced here to illustrate the relationship between agent-level motion and continuum density evolution, which motivates the density-based control laws developed in the remainder of this chapter.

3.2 Problem Formulation

Consider a swarm of N agents with states $\mathbf{x}_i(t) \in \mathcal{R}$. Under deterministic velocity-driven motion, the the constraint-free agent dynamics are given by

$$\dot{\mathbf{x}}_i(t) = \mathbf{v}(t, \mathbf{x}_i(t)). \tag{3.5}$$

The associated swarm density $f_{\mathcal{R}}(t, \mathbf{x})$ evolves according to the continuity equation

$$\frac{\partial f_{\mathcal{R}}(t, \mathbf{x})}{\partial t} = -\nabla \cdot \big(\mathbf{v}(t, \mathbf{x})\, f_{\mathcal{R}}(t, \mathbf{x})\big), \tag{3.6}$$

which is obtained from the Fokker-Planck equation (3.4) under vanishing diffusion, i.e., $\sigma(t, \mathbf{x}) = \mathbf{0}$.

The swarm density is interpreted as a continuum representation of the agent ensemble, where each agent follows (3.5). The objective of this chapter is to synthesize a decentralized velocity field $\mathbf{v}(t, \mathbf{x})$ such that the swarm density $f_{\mathcal{R}}(t, \mathbf{x})$ converges to a desired continuously differentiable target density $f^d_{\mathcal{R}}(\mathbf{x})$, i.e.,

$$\lim_{t\to\infty} f_{\mathcal{R}}(t,\mathbf{x}) = f^d_{\mathcal{R}}(\mathbf{x}), \quad \forall \mathbf{x} \in \mathcal{R}. \tag{3.7}$$

The control law is required to be implementable in a decentralized manner using only local information, motivating the density estimation and velocity synthesis methods developed in the following sections.

3.2.1 Problem Definition

Let $\mathcal{S}(t) = \{\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_N(t)\}$ be a homogeneous swarm of N agents that are distributed over configuration space i.e. $\mathbf{x}_i(t) \in \mathcal{R}\ \forall i \in \{1,\ldots,N\}$ with the initial swarm density distribution $f_{\mathcal{R}}(t_0,\mathbf{x})$. For a given continuously differentiable desired density distribution $f^d_{\mathcal{R}}(\mathbf{x}) \in C^1(\mathcal{R},\mathbb{R})$, the *density control problem* is defined as synthesizing a local density feedback law based on *velocity field* $\mathbf{v}(t,\mathbf{x})$ such that, when followed by agents, the swarm density distribution satisfies the following condition,

$$\lim_{t\to\infty} f_{\mathcal{R}}(t,\mathbf{x}) = f^d_{\mathcal{R}}(\mathbf{x}). \tag{3.8}$$

As it is illustrated in Figure 3.1, the idea is that, the sequence of desired density commands (analogous to reference signals in feedback control systems) can be generated using the density guidance method presented in Chapter 2 or any other density guidance method (see [37]), and the density feedback law to be introduced in this chapter can be used to track these desired density commands. Along with the information of current estimated density $\hat{f}_{\mathcal{R}}(t,\mathbf{x})$, on-board controllers will translate density commands $f^d_{\mathcal{R}}(\mathbf{x})$ to a velocity field $\mathbf{v}\big(t,\mathbf{x}(t)\big)$ in a decentralized feedback fashion for each agent to track. The resulting velocity field will be a continuous function over $\mathcal{R}$ which empirically reduces collision propensity and motivates the collision analysis developed in Chapter 4.

The key desired property in density feedback control architecture is the *decentralization*

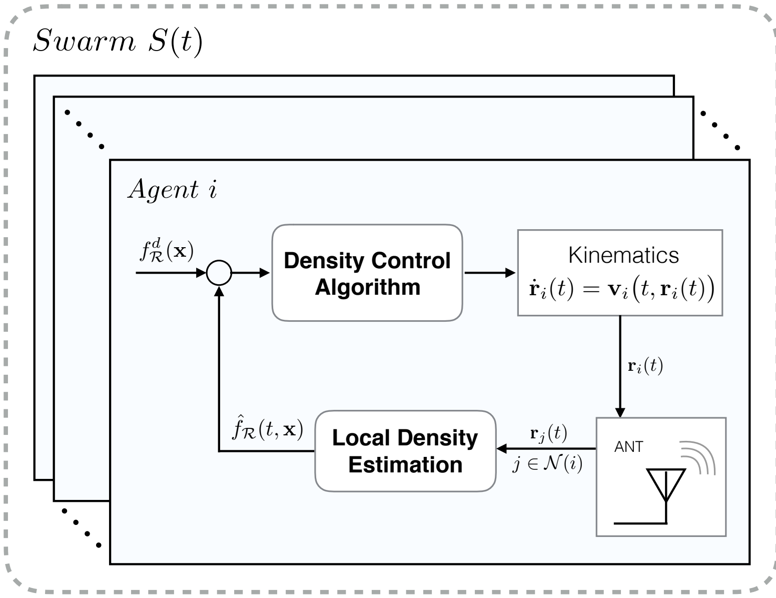

Figure 3.1: The block diagram of the local density feedback control architecture for swarm of agents to track desired density commands.

of computations and relying on local resources on each agent. Therefore, both density estimation and density control algorithm needs to be able to operate on local information. These two algorithmic components will be introduced next.

3.3 Local Density Estimation

As it can be seen from Figure 3.1, the feedback law to generate velocity field requires estimates of the *local density distribution*, i.e., density distribution around each agent and its gradient. To that end, before getting into the details of velocity field generation, this section addresses the local density estimation problem for the swarm of agents.

Having finite number of agents, N, prohibits the exact calculation of the local densities $\rho(t, \mathbf{x}_i(t))$, $\forall \mathbf{x}_i(t) \in S(t)$. Therefore, we aim to estimate swarm density distribution $f_{\mathcal{R}}(t, \mathbf{x})$, or equivalently, approximate $\rho(t, \mathbf{x})$ from given agent configurations. Previous work [44] utilized simple counting algorithm, which partitions the configuration space into subspaces, then calculates the density profile by just counting the number of agents within each subspace (see Figure 3.2). Due to its procedure, this methodology resembles Eulerian approach for numerical solutions of partial differential equations. However, this approach not only renders the algorithm centralized, but also the local estimations will be sensitive to the partitioning method, as partitions affect the resolution of the density profile over the configuration space. Furthermore, sole counting approach does not scale well with the number of partitions from an algorithmic standpoint, hence, preferred increase in the resolution of density profile induces a bottleneck via computational complexity.

In order to obtain decentralized and domain independent estimation, we shifted our focus to Lagrangian approaches in which all the calculations are performed around a neighborhood of particles. The notion of particles aligns well with each individual agent in the swarm, and the ability to perform the calculations from the perspective of each agent enables local computations, hence, decentralization of the coordination algorithm. The first naive attempt to implement this approach can be simple Lagrangian counting algorithm, i.e., each agent counts the number of agents within the prescribed vicinity. Though this naive method meets the decentralization requirement, it fails to capture how close the agents are within the neighborhood, which can be a critical piece of information as far as collision avoidance is

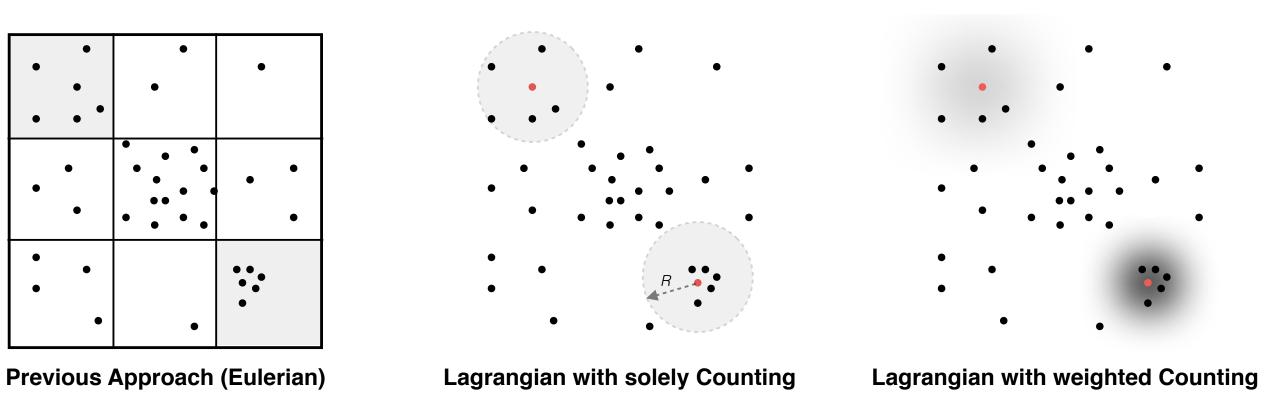

Figure 3.2: Different Estimation Methods: Solely counting Eulerian approach (left), solely counting Lagrangian approach (middle), weighted counting Lagrangian approach (right).

concerned. Furthermore, the density information is discretized in a non-smooth way, i.e., small changes in agent locations, changes particularly close to vicinity border, can affect the propagation of density substantially. In order to reflect the notion of closeness to the density calculations more structure is required in the estimation process that exploits measure of distance.

Obtaining the density information from particles/samples has been studied in different domains such as *Kernel Density Estimation* (KDE) [113, 102] for estimating the probability density function of a random variable from a given set of samples, and *Smooth Particle Hydrodynamics* (SPH) [94] for approximating the flow density in numerical solutions of the partial differential equations for fluid dynamics. Following [102], the kernel density estimate of $f_{\mathcal{R}}(t, \mathbf{x})$ for any point $\mathbf{x} \in \mathcal{R}$ at time t is given by,

$$\hat{f}_{\mathcal{R}}(t, \mathbf{x}) = \int_{\mathcal{R}} \frac{1}{\det(\mathbf{H})} K\Big(\mathbf{H}^{-1}(\mathbf{x} - \xi)\Big) \, \mathrm{d}P_N(t, \xi) \tag{3.9}$$

where $\mathrm{d}P_N(t, \xi)$ is defined as,

$$\mathrm{d}P_N(t, \xi) = \frac{1}{N} \sum_{i=1}^{N} \delta\big(\xi - \mathbf{x}_i(t)\big) \, \mathrm{d}\xi,$$

in which $\delta(\cdot)$ stands for the multi-dimensional Dirac δ-function defined as, $\delta(\xi - \mathbf{x}) = \prod_{k=1}^{d} \delta\big(\xi_k - [x]_k\big)$, and substitution reduces (3.9) to the following form

$$\hat{f}_{\mathcal{R}}(t, \mathbf{x}) = \frac{1}{N \det(\mathbf{H})} \sum_{i=1}^{N} K\Big(\mathbf{H}^{-1}\big(\mathbf{x} - \mathbf{x}_i(t)\big)\Big). \tag{3.10}$$

In (3.10), $K(\cdot) : \mathbb{R}^d \to \mathbb{R}$ is called a multivariate *smoothing kernel* function with $d \times d$ symmetric and positive definite *smoothing parameter* matrix $\mathbf{H}$. Generally, $\mathbf{H}$ is chosen as a diagonal matrix $\mathrm{diag}(h_1, h_2, \ldots, h_d)$ with idential entries in each dimension, i.e., $h_k = h, \ \forall k \in \{1, \ldots, d\}$ [68], where h is known as *interpolation length*. Also, a multivariate kernel typically takes the following product form (e.g., Gaussian kernel function), $K(\mathbf{x}) = K([x]_1)K([x]_2)\ldots K([x]_d)$. Consequently, the local density distribution around an agent i is estimated as follows,

$$\hat{f}_{\mathcal{R}}(t, \mathbf{x}_i(t)) = \frac{1}{Nh^d} \sum_{j=1}^{N} \left[\prod_{k=1}^{d} K\left(\frac{\left[\mathbf{x}_i(t) - \mathbf{x}_j(t)\right]_k}{h} \right) \right]. \tag{3.11}$$

As (3.11) illustrates, estimation is performed locally from each agent's perspective, i.e., each agent uses the measurements of relative distances of other agents. The choice of kernel function and its parameters plays a significant role in the characteristic of the local density estimates as it will be explained next.

3.3.1 Smoothing Kernel Function

Kernel density estimation boils down to a proper choice of kernel function $K(\cdot) : \mathbb{R} \to \mathbb{R}$ and smoothing parameter h. Typically, kernel functions are chosen such that they satisfy the following properties [102],

$$\sup_{-\infty < x < \infty} \left|K(x)\right| < \infty, \tag{3.12a}$$

$$\int_{-\infty}^{\infty} \left|K(x)\right| \, dx < \infty, \tag{3.12b}$$

$$\lim_{x \to \infty} \left|xK(x)\right| = 0. \tag{3.12c}$$

The following theorem [19] states the set of conditions for probability density function estimation (3.11) to be asymptotically unbiased in the sense that if smoothing parameter $h = h(N)$ is chosen as function of N such that

$$\lim_{N \to \infty} h(N) = 0. \tag{3.13}$$

Theorem 2 *Suppose $K(x)$ is a* Borel function *satisfying the conditions (3.12). Let $g(x)$ satisfy $\int_{-\infty}^{\infty} |g(x)|\, dx < \infty$ and let $h(N)$ be a sequence of positive constants satisfying (3.13). Define*

$$\hat{g}(x) = \frac{1}{h(N)} \int_{-\infty}^{\infty} K\left(\frac{\xi}{h(N)}\right) g(x-\xi)\ \mathrm{d}\xi, \tag{3.14}$$

then the following relation holds for every point x

$$\lim_{N\to\infty} \hat{g}(x) = g(x) \int_{-\infty}^{\infty} K(\xi)\ \mathrm{d}\xi. \tag{3.15}$$

Consequently, the estimates defined by (3.10) are asymptotically unbiased *at all points $\mathbf{x}$ at which $f_{\mathcal{R}}(t, \mathbf{x})$ is continuous if the function $K(x)$ satisfies (3.12) and if h satisfies (3.13) along with the following,*

$$\int_{-\infty}^{\infty} K(\xi)\ \mathrm{d}\xi = 1. \tag{3.16}$$

Definition 5 *A kernel function $K(\cdot) : \mathbb{R}^d \to \mathbb{R}$ called* radially symmetric *if there exists a function $\hat{K}(\cdot) : [0, \infty) \to \mathbb{R}$ such that $K(\mathbf{x}) = \hat{K}\left(\|\mathbf{x}\|_2\right)\ \forall \mathbf{x} \in \mathbb{R}^d$. Also, a function is radially symmetric if and only if $K(O\mathbf{x}) = K(\mathbf{x})$ for any orthogonal matrix $O \in \mathbb{R}^{d\times d}$.*

In the context of this work, $K(\mathbf{x})$ is chosen to be *radially symmetric* and *unimodal*, i.e., $K(\mathbf{x})$ has a single maximum/peak (also known as *mode*) that occurs at $\mathbf{x} = \mathbf{0}$. Also note that the definition (3.14) with (3.13),(3.15) and (3.16), implies that

$$\lim_{N\to\infty} \left(\frac{1}{h(N)} K\left(\frac{\xi}{h(N)}\right)\right) = \delta(\xi).$$

In other words, as $h(N) \to 0$ the smoothing kernel approaches to Dirac δ-function, hence, the estimation $\hat{f}_{\mathcal{R}}(t, \mathbf{x})$ approaches to the continuous actual swarm density distribution $f_{\mathcal{R}}(t, \mathbf{x})$ as given by the Theorem 1 in [129]:

Theorem 3 (Weak consistency) *Let the assumption $\lim_{N\to\infty} Nh(N) = \infty$ hold. Then, at each point of continuity x of f, the estimator $f_N(x)$ is weakly consistent, i.e. for each*

$\epsilon > 0$

$$\lim_{N\to\infty} \mathbb{P}(|f_N(x) - f(x)| > \epsilon) = 0. \tag{3.17}$$

Note that, for finite N, smoothing parameter h can not be arbitrary small, as after certain h_{min} the estimate $\hat{f}_{\mathcal{R}}(t, \mathbf{x})$ becomes nothing but N number of impulses at agent locations.

The kernel function $K(x)$ determines the characteristics of the spikes in the estimation $\hat{f}_{\mathcal{R}}(t, \mathbf{x})$ and this estimate is continuously differentiable, i.e. $\hat{f}_{\mathcal{R}}(t, \mathbf{x}) \in C^1(\mathcal{R}, \mathbb{R})$ provided that $K(x) \in C^1(\mathcal{R}, \mathbb{R})$. Also observe that, if $K(x) \geq 0 \;\; \forall x \in \mathcal{R}$ and $K(x)$ satisfies (3.16), then the density estimates given by (3.11) satisfy the properties of probability density functions given as (3.2). Figure 3.3 illustrates a density estimation example with high number of agents randomly dispersed within triangular bins with respect to a density field over an arbitrary domain using *Gaussian kernel function*. The choice of kernel functions can depend various factors such as desired density features and calculation complexity for practical implementations. Table 3.3.1 presents various examples of kernel functions [102] with their Fourier transforms and roughness values.

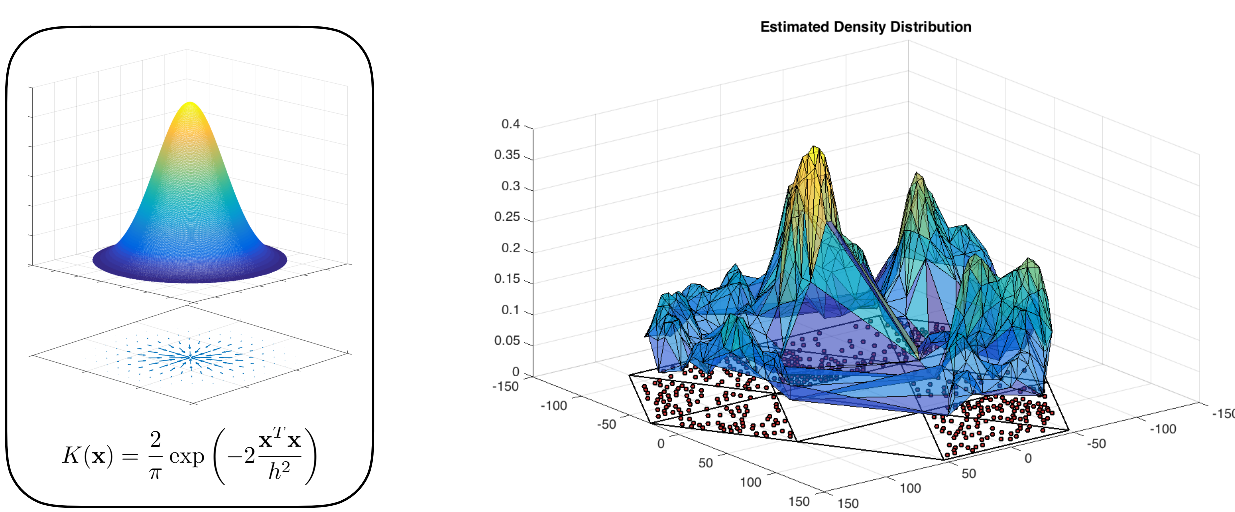

Figure 3.3: An Estimation Example: Gaussian kernel function with its gradient (left), and the estimation performed at agent locations for $N = 500$ with the 2-dimensional Gaussian kernel function where h = 75 (right).

Table 3.1: Examples of Kernel Functions with Fourier Transform and Roughness Parameters

$K(x)$	$\int_{-\infty}^{\infty} e^{i\omega x}\, K(x)\, dx$	$\int_{-\infty}^{\infty} K^2(x)\, dx$
$\frac{1}{2},\quad \lvert x\rvert \le 1$ $0,\quad \lvert x\rvert > 1$	$\dfrac{\sin\,\omega}{\omega}$	$\dfrac{1}{2}$
$1-\lvert x\rvert,\quad \lvert x\rvert \le 1$ $0,\quad \lvert x\rvert > 1$	$\left\{\dfrac{\sin\,(\omega/2)}{\omega/2}\right\}^2$	$\dfrac{2}{3}$
$\frac{4}{3}-8x^2+8\lvert x\rvert^3,\quad \lvert x\rvert < \frac{1}{2}$ $\frac{8}{3}\big(1-\lvert x\rvert\big)^3,\quad \frac{1}{2} \le \lvert x\rvert \le 1$ $0,\quad \lvert x\rvert > 1$	$\left\{\dfrac{\sin\,(\omega/4)}{\omega/4}\right\}^4$	0.96
$\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}$	$e^{-\frac{1}{2}\omega^2}$	$\dfrac{1}{2\sqrt{\pi}}$
$\frac{1}{2}e^{-\lvert x\rvert}$	$\big(1+\omega^2\big)^{-1}$	$\dfrac{1}{2}$
$\dfrac{1}{\pi}\big(1+x^2\big)^{-1}$	$e^{-\lvert\omega\rvert}$	$\dfrac{1}{\pi}$
$\dfrac{1}{2\pi}\left\{\dfrac{\sin\,(x/2)}{x/2}\right\}^2$	$1-\lvert\omega\rvert,\quad \lvert\omega\rvert \le 1$ $0,\quad \lvert\omega\rvert > 1$	$\dfrac{1}{3\pi}$

Proposition 1 *Density estimations defined by (3.10) with positive unimodal kernel functions are bounded at agent locations as follows:*

$$\frac{1}{Nh^d}K\big(\mathbf{0}\big) \;\le\; \hat{f}_{\mathcal{R}}\big(\mathbf{x}_i(t)\big) \;\le\; \frac{1}{h^d}K\big(\mathbf{0}\big).$$

Proof: The lower bound follows from Condition 3.12c and substituting $\mathbf{x} = \mathbf{x}_i$ along with $\|\mathbf{x}_i - \mathbf{x}_j\| \to \infty$ for all $i \neq j$ in (3.10). The upper bound follows from using unimodality of $K(\mathbf{x})$ and substituting $\mathbf{x} = \mathbf{x}_i$ along with $\mathbf{x}_j = \mathbf{x}_i\ \forall j$. ■

The self-contribution of each particle provides a strictly positive lower bound on the density estimate as it is given by Proposition 1.

3.3.2 Selection of Smoothing Parameter

The smoothing parameter controls the degree of smoothing in the estimation process, and the necessity of $h > h_{min}$ (or finite N) incurs a bias error for the estimation $\hat{f}_{\mathcal{R}}(t, \mathbf{x})$. Selection of h affects the behavior of the density estimator. More specifically, if h is small, then the estimator will give results with high noise and variance (more spiky estimates), similarly if h is large then the estimations will be too smooth, and perform poorly to capture the characteristics of the actual density. There are various empirical methods to pick smoothing parameter h, and the choice of the method is strongly tied with the specific application at hand, but typically, smoothing parameter h is chosen such that it minimizes the overall integral of mean-squared error (MSE) between $\hat{f}_{\mathcal{R}}(t, \mathbf{x})$ and $f_{\mathcal{R}}(t, \mathbf{x})$ which is defined as

$$\mathbb{E}\Big[\big(\hat{f}_{\mathcal{R}}(t, \mathbf{x}) - f_{\mathcal{R}}(t, \mathbf{x})\big)^2\Big] \;=\; \text{bias}\big(\hat{f}_{\mathcal{R}}(t, \mathbf{x})\big)^2 \;+\; \text{var}\big(\hat{f}_{\mathcal{R}}(t, \mathbf{x})\big). \tag{3.18}$$

Note that the expression (3.18) demonstrates the *bias-variance trade-off* where the bias and variance expressions are as follows [68],

$$\text{bias}\big(\hat{f}_{\mathcal{R}}(t, \mathbf{x})\big) \;=\; \mathbb{E}\Big[\hat{f}_{\mathcal{R}}(t, \mathbf{x}) - f_{\mathcal{R}}(t, \mathbf{x})\Big] \;=\; h^{\nu}\frac{\kappa_{\nu}}{\nu!}\sum_{k=1}^{d}\frac{\partial^{\nu}}{\partial x_k^{\nu}} f_{\mathcal{R}}(t, \mathbf{x}) + o(h^{\nu}), \tag{3.19}$$

$$\text{var}\big(\hat{f}_{\mathcal{R}}(t, \mathbf{x})\big) \;=\; \mathbb{E}\big[\hat{f}_{\mathcal{R}}(t, \mathbf{x})^2\big] - \mathbb{E}\big[\hat{f}_{\mathcal{R}}(t, \mathbf{x})\big]^2 \;=\; \frac{f_{\mathcal{R}}(t, \mathbf{x})R(K)^d}{Nh^d} + O\Big(\frac{1}{N}\Big), \tag{3.20}$$

where ν is the *order* of $K(x)$ which is the index of first non-zero *moment* κ_j:

$$\kappa_j = \int_{-\infty}^{\infty} x^j K(x)\ \mathrm{d}x,$$

and $R(K)$ is called *roughness* of $K(x)$ and calculated as,

$$R(K) = \int_{-\infty}^{\infty} K^2(x)\ \mathrm{d}x.$$

In (3.19) the expression $o(h^\nu)$ means the remaining terms are of smaller order than h^ν as $h \to \infty$, also it is assumed that $\frac{\partial^{\nu+1}}{\partial x_k^{\nu+1}} f_{\mathcal{R}}(t, \mathbf{x})$ exists. In (3.20) the expression $O(1/N)$ means the remaining terms are of the same order as $1/N$.

$R(K)$ is used as the measure of difficulty to estimate $K(x)$, i.e., larger the value of $R(K)$ more difficult it is to estimate $K(x)$ [118]. Unfortunately, closed-form solution for h that achieves the minimum for the integral of (3.18) does not exists [68]. A rule of thumb for determining h is to replace $f_{\mathcal{R}}(t, \mathbf{x})$ in (3.19) and (3.20) with another elementary function (e.g., Gaussian) and find the minimizing h as if the actual density $f_{\mathcal{R}}(t, \mathbf{x})$ is the elementary function. The general form for minimizing h is given as follows,

$$h^* = \hat{\sigma}\ C_\nu(K, d)\ N^{-1/(2\nu+d)}, \tag{3.21}$$

where $\hat{\sigma}$ is the standard deviation of the N sample, and $C_\nu(K, d)$ is a constant depends on the order of K, number of dimensions d and the roughness of the elementary function that is assumed to be the actual density. Numerical $C_\nu(K, d)$ values for some elementary functions are given in Table 7 in [68]. The further results in this research reveal that the choice of smoothing parameter h has a direct implication on the transient behavior of the motion of agents and collision radius for each agent, more specifically, an arbitrary collision radius can be enforced via choosing proper smoothing parameter in the density estimation process.

Remark 4 *As it can be seen from (3.19), for symmetric non-negative kernels, i.e.,* second-order *kernels ($\nu = 2$), the bias error is $O(h^2)$. However, as it is mentioned in [94], this error will be smaller when particles being propagated by certain dynamical equations that drive the particles to lower potential energy states.*

3.4 Density Control via Velocity Field Generation

The purpose of the velocity field generation algorithm is to relate macroscopic level signals, i.e., swarm density distribution to agent level dynamical quantities, i.e., velocity of the vehicle. In the derivation of velocity field synthesis method presented in this research, partial differential equations, more specifically the heat equation plays a critical role.

The heat equation is a parabolic partial differential equation that describes the evolution of temperature in time over a region:

$$\frac{\partial u(t, \mathbf{x})}{\partial t} = \alpha \, \Delta u(t, \mathbf{x}), \tag{3.22}$$

where $\alpha > 0$ is called thermal diffusivity constant. This equation is also known as a special form of *diffusion equation* in which the *diffusion constant* α is considered to be a function of $\mathbf{x}$ and u. The approach is to employ heat equation to describe the evolution of the swarm density distribution given by $f_{\mathcal{R}}(t, \mathbf{x})$, in time.

Similar to feedback control law logic, let the difference between the current and desired probability density functions be

$$\Phi(t, \mathbf{x}) := f_{\mathcal{R}}(t, \mathbf{x}) - f_{\mathcal{R}}^{d}(\mathbf{x}) \qquad \mathbf{x} \in \mathcal{R}$$

The idea is to find the velocity field that transforms Equation (3.6) to heat equation given in (3.24a). For this purpose, the following feedback law is proposed to compute the velocity as a function of $\Phi(t, \mathbf{x})$ and $f_{\mathcal{R}}(t, \mathbf{x})$,

$$\mathbf{v}(t, \mathbf{x}) = -\alpha \, \frac{\nabla \Phi(t, \mathbf{x})}{f_{\mathcal{R}}(t, \mathbf{x})}. \tag{3.23}$$

Noting that the desired density is constant in time, i.e. $\partial f_{\mathcal{R}}^{d}(\mathbf{x})/\partial t = 0$, we can rewrite (3.6) with the proposed velocity field as follows,

$$\frac{\partial \Phi}{\partial t}(t, \mathbf{x}) = \alpha \, \Delta \Phi(t, \mathbf{x}), \quad \mathbf{x} \in \mathcal{R}, \; t > t_0 \tag{3.24a}$$

$$\Phi(t_0, \mathbf{x}) = f_{\mathcal{R}}(t_0, \mathbf{x}) - f^d_{\mathcal{R}}(\mathbf{x}). \tag{3.24b}$$

and suppose that $\Phi(t, \mathbf{x})$ satisfies the boundary condition $\nabla \Phi(t, \mathbf{x}) = 0$ on $\partial \mathcal{R}$. Then $\int_{\mathcal{R}} \Phi(t, \mathbf{x}) \, \mathrm{d}\mathbf{x}$ is conserved for all $t \geq t_0$, and, since we have,

$$\int_{\mathcal{R}} f_{\mathcal{R}}(t, \mathbf{x}) \, \mathrm{d}\mathbf{x} = \int_{\mathcal{R}} f^d_{\mathcal{R}}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \quad \forall t \geq t_0 \tag{3.25}$$

the following holds,

$$\int_{\mathcal{R}} \Phi(t, \mathbf{x}) \; \mathrm{d}\mathbf{x} = 0 \quad \forall t \geq t_0. \tag{3.26}$$

Intuitively, the sink/source free diffusion process (heat equation) converges to its average value, i.e. achieves consensus over the process domain. Defining the diffusion process over $\Phi(t, \mathbf{x})$ i.e. the profile of difference from desired density, will drive $\Phi(t, \mathbf{x})$ to its average value across the domain which is zero. Therefore the diffusion process will ensure convergence to the desired density, since $\Phi(t, \mathbf{x}) \to 0 \; \forall \mathbf{x} \in \mathcal{R}$ is equivalent to $f_{\mathcal{R}}(t_0, \mathbf{x}) \to f^d_{\mathcal{R}}(\mathbf{x})$. To this end, the following theorem gives the main result which shows that, in case of $N \to \infty$, the swarm converges to the desired density distribution from any initial distribution when the velocity field is synthesized with (3.23). Also recall from Theorem 3 that, in the continuum limit $N \to \infty$, density estimates converge to the true density distribution, i.e., $\hat{f}_{\mathcal{R}} \to f_{\mathcal{R}}$, with probability 1.

Theorem 4 *Consider a swarm $\mathcal{S}(t)$ as a continuum, commanded with continuous desired density function $f^d_{\mathcal{R}}(\mathbf{x})$, where the motion of each agent is governed by the following dynamics,*

$$\dot{\mathbf{x}}(t) = -\alpha \, \frac{\nabla \Phi(t, \mathbf{x})}{f_{\mathcal{R}}(t, \mathbf{x})}. \tag{3.27}$$

As $t \to \infty$, for any initial conditions of the agents in the swarm, the density difference $\Phi(t, \mathbf{x})$

for each agent asymptotically converges to zero i.e.

$$\lim_{t\to\infty} \Phi(t,\mathbf{x}) = 0 \quad \mathbf{x} \in \mathcal{R},$$

hence the swarm density distribution $f_{\mathcal{R}}(t,\mathbf{x})$ *approaches to the desired density distribution* $f^d_{\mathcal{R}}(\mathbf{x})$*, and velocities of all the agents vanish asymptotically, i.e.,* $\lim_{t\to\infty} \dot{\mathbf{x}}(t) = 0$.

Proof: For the analysis we define the following positive definite function as the Lyapunov function for the swarm:

$$V(t) = \frac{1}{2}\int_{\mathcal{R}} \left(\frac{f_{\mathcal{R}}(t,\mathbf{x})}{\alpha}\right)^2 \dot{\mathbf{x}}^{\mathsf{T}}\dot{\mathbf{x}}\ \mathrm{d}\mathbf{x} \ = \ \frac{1}{2}\int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})^{\mathsf{T}}\nabla\Phi(t,\mathbf{x})\ \mathrm{d}\mathbf{x}. \tag{3.28}$$

Taking the time derivative of Lyapunov function $V(t)$ yields,

$$\dot{V}(t) = \int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})^{\mathsf{T}}\nabla\dot{\Phi}(t,\mathbf{x})\ \mathrm{d}\mathbf{x},$$

by using heat equation $\dot{\Phi}(t,\mathbf{x}) = \alpha\Delta\Phi(t,\mathbf{x})$ and the third derivative vector calculus,

$$\dot{V}(t) = \int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})^{\mathsf{T}}\Big(\nabla\big(\alpha\Delta\Phi(t,\mathbf{x})\big)\Big)\mathrm{d}\mathbf{x} \ = \ \alpha\int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})^{\mathsf{T}}\Delta\big(\nabla\Phi(t,\mathbf{x})\big)\ \mathrm{d}\mathbf{x}.$$

Employing a change of variables as $\zeta(t,\mathbf{x}) = \nabla\Phi(t,\mathbf{x})$ yields

$$\dot{V}(t) = \alpha\int_{\mathcal{R}} \zeta(t,\mathbf{x})^{\mathsf{T}}\Delta\zeta(t,\mathbf{x})\ \mathrm{d}\mathbf{x}$$

Also the boundary condition $\nabla\Phi(t,\mathbf{x}) = 0$ on $\partial\mathcal{R}$ can be written as *Dirichlet* boundary condition of $\zeta(t,\mathbf{x}) = 0$ on $\partial\mathcal{R}$. Then we can conclude $\dot{V}(t) < 0$ because of the fact that, *Dirichlet* eigenvalue problem for Δ operator has countably many strictly negative eigenvalues [88]. Consequently, $\lim_{t\to\infty} \nabla\Phi(t,\mathbf{x}) = 0$, which implies $\lim_{t\to\infty} \dot{\mathbf{x}}(t) = 0$. Furthermore,

$$\lim_{t\to\infty} \Phi(t,\mathbf{x}) = \text{const.} \tag{3.29}$$

Since $f_{\mathcal{R}}(t,\mathbf{x})$ and $f^d_{\mathcal{R}}(\mathbf{x})$ are continuous then $\Phi(t,\mathbf{x})$ is also continuous, and because of (3.26), $\lim_{t\to\infty}\Phi(t,\mathbf{x}) = 0$, which implies asymptotic convergence

$$\lim_{t\to\infty} f_{\mathcal{R}}(t,\mathbf{x}) = f^d_{\mathcal{R}}(\mathbf{x}) \qquad \forall \mathbf{x}\in\mathcal{R}.$$

$\blacksquare$

Remark 5 *The result (3.29) implies that even if the relation (3.25) does not hold, the desired density $f^d_{\mathcal{R}}(\mathbf{x})$ will be achieved with a bias of $\Phi(t,\mathbf{x}) = \text{const}$. This means if the amount of agents over the domain $\mathcal{R}$ increase or decrease at an instant without tuning N in (3.11), the velocity field will act to preserve the characteristics of the desired density $f^d_{\mathcal{R}}(\mathbf{x})$.*

In the following years the results in Theorem 4 has been extended [135, 134] in two aspects. First the velocity field is generalized such that it captures the stochastic effects/motions modeled by (3.3) with the assumption that Wiener process is a full state vector, i.e., $\mathbf{W}(t)\in\mathbb{R}^d$ and diffusion function is a scalar field i.e., $\sigma(t,\mathbf{x})\in\mathbb{R}$. Second, Lyapunov-based stability analysis has been performed to show a stronger result of *exponential stability* as given by the following theorem proof of which is provided in Appendix B.1.

Theorem 5 *[135, 134] Consider the following velocity field for the swarm $\mathcal{S}(t)$,*

$$\dot{\mathbf{x}}(t) = -\frac{\alpha(t,\mathbf{x})\nabla\Phi(t,\mathbf{x}) - \nabla\big(\sigma^2(t,\mathbf{x})\, f_{\mathcal{R}}(t,\mathbf{x})/2\big)}{f_{\mathcal{R}}(t,\mathbf{x})}. \tag{3.30}$$

where $\alpha(t,\mathbf{x})>0$ is the diffusion parameter that satisfies,

$$\sup_{(t,\mathbf{x})}\ \alpha(t,\mathbf{x}) < \infty \qquad \text{and} \qquad \inf_{(t,\mathbf{x})}\ \alpha(t,\mathbf{x}) > 0.$$

If the solution satisfies $f_{\mathcal{R}}(t,\mathbf{x})>0$ for all $t>0$, then $\|\Phi(t,\mathbf{x})\|_{L^2(\mathcal{R})}\to 0$ exponentially.

The velocity field requires the local density distribution and its gradient, and this information is estimated on-board of each agent with the local information, hence, making the algorithm decentralized. By each agent following this local velocity accurately with the feedback of local density estimate, the swarm will be driven to the desired density $f^d_{\mathcal{R}}(\mathbf{x})$ autonomously. Also, as it is mentioned in Remark 4, since the heat equation, and hence, the velocity field spreads agents locally uniformly, the bias error will be better than $O(h^2)$.

The velocity field defined by (3.23) represents a non-linear single integrator dynamics and does not account for any sophisticated agent dynamics. Although this can be perceived as a constraining assumption, any real-world application of the proposed control law can be implemented as a velocity tracking application as it is demonstrated in Section 3.5.2. However, the control commands should comply with each agent's performance envelope, i.e. the maximum rate of change in the state. The next result show how this upper bound can be set by proper choice of estimation parameters and control gain α.

Lemma 4 *Given a swarm of agents $\mathcal{S}(t) = \{\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_N(t)\}$, $\mathbf{x}_i(t) \in \mathbb{R}^d \; \forall i$, where, (i) the motion of each is governed by (3.23), (ii) agents perform kernel density estimation using (3.10), (iii) both the unimodal kernel function $K(\mathbf{x})$ and the desired density commands $f^d_{\mathcal{R}}(\mathbf{x})$ are Lipschitz continuous with constants $L_K > 0$ and $L_{\rho_D} \geq 0$ respectively, then for the velocity field the following holds:*

$$\|\dot{\mathbf{x}}_i\|_2 \;\leq\; \frac{\alpha}{K_0}\,\big((N-1)L_K + Nh^d L_{\rho_D}\big),$$

where $K_0 = K\big(\mathbf{0}\big)$.

Proof: Observing the following,

$$\|\dot{\mathbf{x}}_i\|_2 \;=\; \alpha\,\frac{\left\|\nabla \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i) - \nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2}{\hat{f}_{\mathcal{R}}(t,\mathbf{x}_i)} \;\leq\; \alpha\,\frac{\max\;\left\|\nabla \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i) - \nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2}{\min\;\hat{f}_{\mathcal{R}}(t,\mathbf{x}_i)},$$

and using Proposition 1 for the lower bound on estimated density one has,

$$
\begin{aligned}
\alpha \frac{\max \ \left\|\nabla \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i) - \nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2}{\min \ \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i)} &= \frac{\alpha N h^d}{K_0} \ \max \ \left\|\nabla \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i) - \nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2 \\
&\leq \frac{\alpha N h^d}{K_0} \left(\max \left\|\nabla \hat{f}_{\mathcal{R}}(t,\mathbf{x}_i)\right\|_2 + \max \left\|\nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2 \right) \\
&\leq \frac{\alpha N h^d}{K_0} \left(\frac{1}{N h^d} \max \sum_{i \neq j} \left\| \nabla K \left(\frac{\mathbf{x}_i - \mathbf{x}_j}{h} \right) \right\|_2 \right. \\
&\qquad \left. + \ \max \left\|\nabla f^d_{\mathcal{R}}(\mathbf{x}_i)\right\|_2 \right) \\
&\leq \frac{\alpha N h^d}{K_0} \left(\frac{N-1}{N h^d} L_K + L_{\rho_D} \right)
\end{aligned}
$$

the last result of Lipschitz constants follows from (473C-c) in [58]. ■

Note that the Lipschitz continuous desired density $f^d_{\mathcal{R}}(\mathbf{x})$ assumption in Lemma 4 may not comply with the output of density guidance algorithm, hence, a solution to the spatially discontinuous desired density command case will be addressed next.

3.4.1 Spatial Smoothing for Desired Density Profiles

The sequence of desired density commands $\theta(k), k \in \mathbb{N}$ generated by the guidance algorithm given in Chapter 2 are piecewise constant over the domain because of the discretization induced by Markov chain formulation. The gradient of the raw guidance commands are zero almost everywhere, and it is not defined at the edges of discretized partitions. In order to be coherent with local density estimation approach, smoothing is applied via spatial convolution of Markov Chain density distribution with the same kernel function used for local density estimation as,

$$
f^d_{\mathcal{R}}(\mathbf{x}) = \rho^d_{MC}(\mathbf{x}) \ * \ K(\mathbf{x}) = \int \rho^d_{MC}(\xi) \ K(\mathbf{x} - \xi) \ \mathrm{d}\xi. \tag{3.31}
$$

where, $\rho^d_{MC}(\cdot) : \mathbb{R}^d \to \mathbb{R}$ is a piecewise constant function over configuration space $\mathcal{R}$ such that,

$$\rho^d_{MC}(\mathbf{x}) := \frac{\theta[i]}{\int_{R_i} \mathrm{d}\mathbf{x}}, \quad \forall \mathbf{x} \in R_i.$$

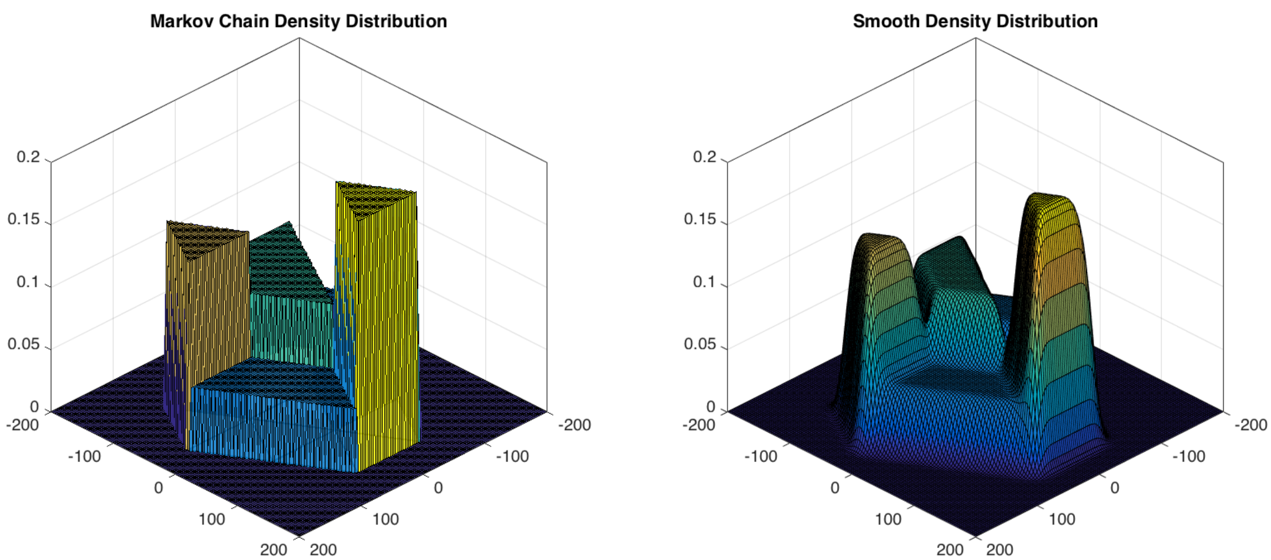

Figure 3.4: Smoothing for the Markov chain Probability Mass Function: Smoothing kernels are applied to the desired probability mass function obtained from Markov chain based density guidance (left), and continuously differentiable desired density profiles (right) are attained to be used in density feedback control.

The effective range of kernel functions, i.e. the range for agents to communicate or sense each other, is generally given by the multiple ν of smoothing parameter h for kernels with compact support, i.e. $\mathcal{K} = \{\mathbf{x} : \|\mathbf{x}\| \leq \nu h\}$, where ν is the order of the kernel. Hence, for a given configuration space $\mathcal{R}$ of the swarm, the density estimates and smooth desired density functions are defined over the set $\mathcal{D}$, which is obtained by the Minkowski sum of $\mathcal{K}$ and $\mathcal{R}$:

$$\mathcal{D} = \mathcal{R} + \mathcal{K} = \{r + k : \ r \in \mathcal{R}, \ k \in \mathcal{K}\}.$$

Upper bound for the magnitude of the velocity field given in Lemma 4 can also be calculated for this case, and before presenting the upper bound for desired density commands in the form of (3.31), the following result (473D-d, g) in [58] will be provided.

Lemma 5 *[58] Suppose that f and g are Lebesgue measurable real-valued functions defined μ-almost everywhere in $\mathbb{R}^d$.*

(i) If f is integrable and g is bounded, Lipschitz and defined everywhere, then $f * \nabla g$ and $\nabla(f * g)$ are defined everywhere and equal, where $f * \nabla g = \left(f * \dfrac{\partial g}{\partial \xi_1}, \ \ldots, \ f * \dfrac{\partial g}{\partial \xi_d} \right)$. Moreover, $f * \nabla g = \nabla(f * g)$ is Lipschitz.

(ii) If f is integrable and $\phi : \mathbb{R}^d \to \mathbb{R}^d$ is a bounded measurable function with components $\phi_1, \ \ldots, \ \phi_d$, and we write $(f * \phi)(\mathbf{x}) = \big((f * \phi_1)(\mathbf{x}), \ \ldots, \ (f * \phi_d)(\mathbf{x})\big)$, then

$$\big\|(f * \phi)(\mathbf{x})\big\| \ \leq \ \big\|f\big\|_1 \sup_{\boldsymbol{y} \in \mathbb{R}^d} \big\|\phi(\boldsymbol{y})\big\| \quad \text{for every } \mathbf{x} \in \mathbb{R}^d.$$

Corollary 3 *Given a swarm of agents $\mathcal{S}(t) = \{\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_N(t)\}, \quad \mathbf{x}_i(t) \in \mathbb{R}^d \ \forall i$, where, (i) the motion of each is governed by (3.23), (ii) agents perform kernel density estimation using (3.10), (iii) unimodal kernel function $K(\mathbf{x})$ is Lipschitz continuous with constants $L_K > 0$, and (iv) the desired density commands $f^d_{\mathcal{R}}(\mathbf{x})$ are defined by (3.31), then the following holds for the velocity field:*

$$\|\dot{\mathbf{x}}_i\|_2 \ \leq \ \frac{\alpha}{K\big(\mathbf{0}\big)} \big(N(h^d + 1) - 1\big) \, L_K.$$

.

Proof: For a normalized density function, or probability density function one has $\displaystyle\int_{-\infty}^{\infty} \rho^d_{MC}(\mathbf{x}) \, \mathrm{d}\mathbf{x} = 1$, i.e., $\|\rho^d_{MC}(\mathbf{x})\|_1 = 1$, also $K(\mathbf{x})$ is Lipschitz, hence, using (i) in Lemma 5 yields,

$$\big\|\nabla f^d_{\mathcal{R}}(\mathbf{x})\big\| \ = \ \big\|\nabla\big(\rho^d_{MC}(\mathbf{x}) \ * \ K(\mathbf{x})\big)\big\| \ = \ \big\|\rho^d_{MC}(\mathbf{x}) \ * \ \nabla K(\mathbf{x})\big\|,$$

which is Lipschitz by (i). Now, using (ii) in Lemma 5 one has the Lipschitz coefficient as,

$$\left\|\rho_{MC}^{d}(\mathbf{x}) \ * \ \nabla K(\mathbf{x})\right\| \ \leq \ \left\|\rho_{MC}^{d}(\mathbf{x})\right\|_{1} \sup_{\mathbf{y}\in\mathbb{R}^{d}} \left\|\nabla K(\mathbf{y})\right\| = \ L_{K}$$

The last equality follows from (473C-c) in [58]. Substituting L_K for L_{ρ_D} in Lemma 4 completes the proof. ■

The bounds given by Lemma 4 and Corollary 3 are useful when synthesizing velocity fields that respect agents' performance envelope. It is also important to note that the Lipschitz coefficient L_K can be regulated by the smoothing parameter h, hence, choice of smoothing parameter for kernel density estimation will impose an upper bound on the magnitude of the velocity field agents are able to follow.

3.5 Numerical Simulations & Experimental Validation

3.5.1 Pattern Generation

In this simulation, the capability of the synthesized controller is illustrated to generate complex patterns. In Figure 3.5, a swarm of N = 1000 agents starting from a random sample of positions with uniform distribution is commanded to desired density defined by the picture of Lenna. The picture file processed to construct a normalized continuous desired swarm density function. The local density estimator utilizes a multi-dimensional Gaussian kernel function:

$$K(\mathbf{x}) = \frac{2}{\pi}\exp\left(-2\frac{\mathbf{x}^{\mathsf{T}}\mathbf{x}}{h^2}\right),$$

and the smoothing parameter h is taken as $h = L/20$ where L is the length of the squared domain. The effective radius around each agent is $R_{\text{eff}} = 2h$. Also, the heat equation based controller has the diffusion constant of $\alpha = 5$. Results show that swarm of 1000 agents almost converged to the desired density distribution after $T = 1000$. Also, agents are locally uniformly distributed and agent locations and inter-agent distances are solely determined by the density command. Figure 3.6 shows the integral of point-wise error over the domain in

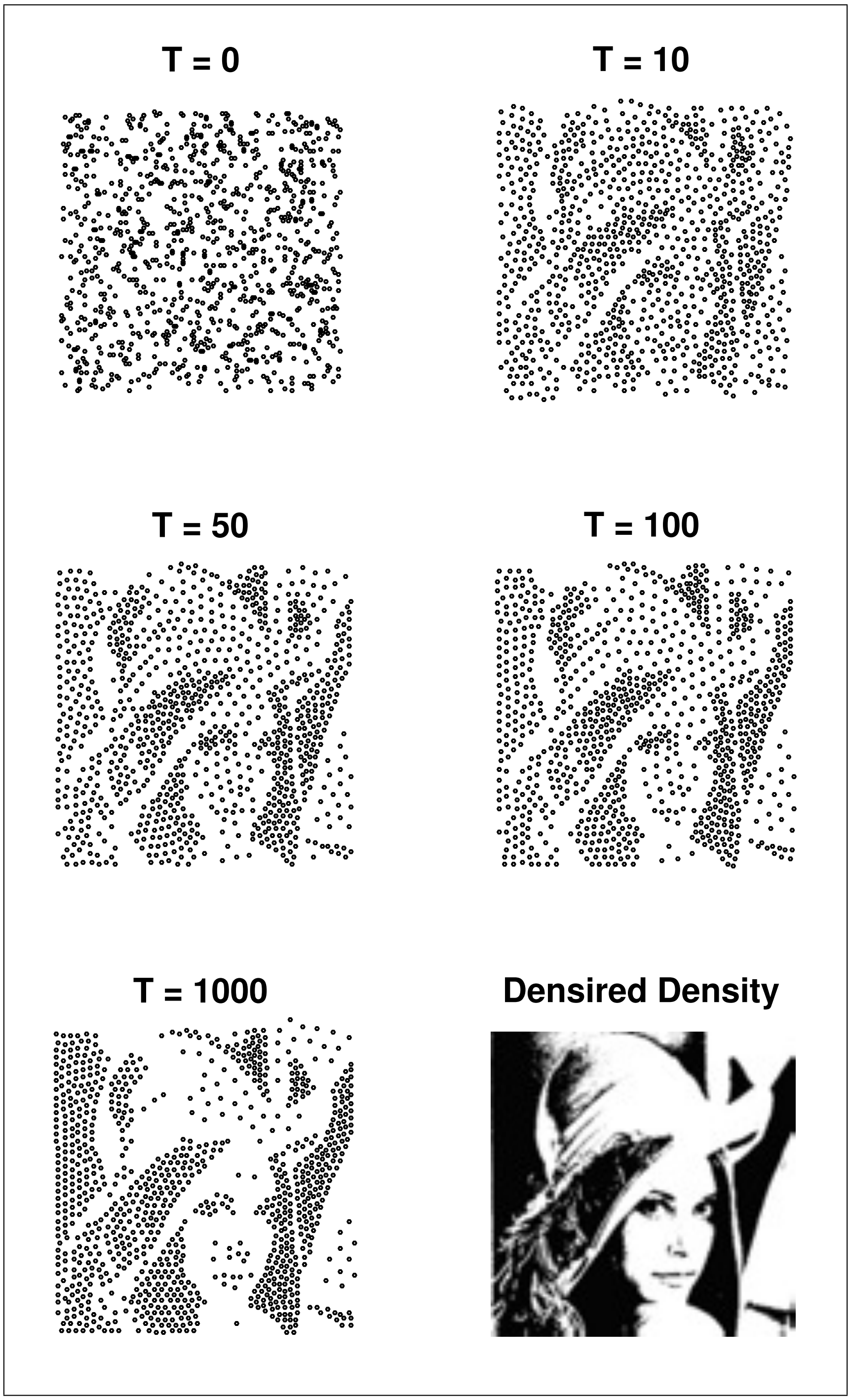

Figure 3.5: The swarm density distribution in time towards desired density.

time defined as,

$$E(t) = \int_{\mathcal{R}} \left| \hat{f}_{\mathcal{R}}(t, \mathbf{x}) - f^d_{\mathcal{R}}(\mathbf{x}) \right| \mathrm{d}\mathbf{x}$$

Even though the error dissipates in time, residual steady-state error arises from finite-

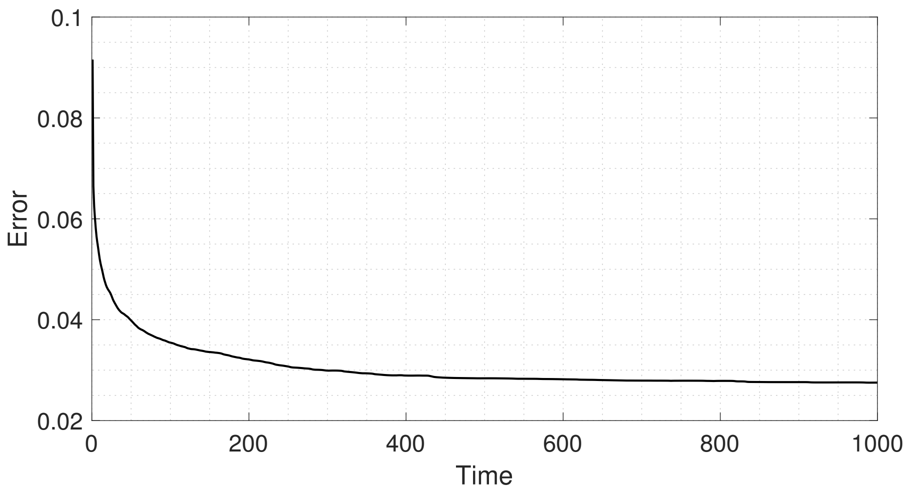

Figure 3.6: The Mean Squared Error in time.

population density estimation and fixed-bandwidth smoothing. h. The reason of this error is the fact that the smoothing parameter is calculated with a heuristic. As it is mentioned earlier, the next task of the research will address dynamically changing smoothing parameters to minimize estimation induced errors.

Low Earth Orbit Density Control

Consider a spacecraft swarm in circular low Earth orbit. The relative dynamics of each spacecraft to the circular orbit are governed by Clohessy-Wiltshire equations [126]:

$$\begin{aligned} \ddot{x} &= 2\omega\dot{y} + 3\omega^2 x + f_x \\ \ddot{y} &= -2\omega\dot{x} + f_y \\ \ddot{z} &= -\omega^2 z + f_z \end{aligned} \tag{3.32}$$

Here x, y and z are the spacecraft's relative position to orbit in local-vertical local-horizontal (LVLH) cartesian coordinates associated with an orbital frame. The x-axis points in the zenith direction, the z-axis is normal to the orbital plane, and the y-axis is determined with right-hand rule. The forces exerted in each axis are given as f_x, f_y and f_z, and ω represents the orbital frequency given as:

$$\omega = \sqrt{\frac{\mu}{a^3}},$$

where a is the radius of the circular orbit and μ is the standard gravitational parameter. For $f_x = f_y = f_z = 0$, the analytical solution of the equations (3.32) reveals that if the initial condition of the swarm satisfies:

$$\dot{y}(0) = -2\omega x(0)$$

then the swarm stays bounded and all spacecraft are period matched. The following definition [65] gives the conditions we assume after this point on.

Definition 6 *A swarm is referred to as a planar Orbit Type Matched (OTM) swarm if it satisfies the following conditions: For all spacecraft* $i = 1, \ldots, N$,

(a) Period Matched with parameter ω, $\dot{y}_i(0) = -2\omega x_i(0)$,

(b) Centroid Matched with parameter y_o, $y_{o,i} := y_i(0) - 2\dot{x}_i(0)/\omega = y_o$

(c) In-plane motion, $z_i(t) = 0 \ \forall t$.

Note that the condition (c) is not necessary for the method proposed in this research, but it is used to keep the discussion contained. When the conditions (a) and (b) are satisfied, all spacecraft follow fuel efficient trajectories that are J_2-invariant elliptical orbits in the LVLH frame. In order to minimize control inputs, these invariant elliptical orbits will be utilized as the steady-state solutions. Therefore, the density commands will be enforced in a different coordinate system that is denoted as Scaled and Rotated (SR) coordinates:

$$\begin{bmatrix} \bar{x}(0) \\ \bar{y}(0) \end{bmatrix} = R^{\mathsf{T}}(\omega t) S^{-1} \begin{bmatrix} x(t) \\ y(t) - y_o \end{bmatrix} \tag{3.33}$$

where,

$$R(\omega t) = \begin{bmatrix} \cos(\omega t) & \sin(\omega t) \\ -\sin(\omega t) & \cos(\omega t) \end{bmatrix} \quad \text{and} \quad S = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}.$$

Note that coordinate transformation is time dependent while the SR coordinates in a planar OTM swarm are time-invariant. Therefore, the density commands will be given in SR coordinates. Moreover, the conditions (a) and (b) implies that the velocity of a spacecraft is defined by the LVLH location of the spacecraft as,

$$\begin{aligned} \dot{x}(t) &= \frac{\omega}{2}\big(y(t) - y_o\big) \\ \dot{y}(t) &= -2\omega x(t), \end{aligned} \tag{3.34}$$

hence, for an OTM swarm, a point in SR coordinates defines a position and velocity in LVLH coordinates. Consequently, the control law (3.23) will be used for achieving the desired density commands in SR coordinates while transitioning the spacecraft between states in LVLH coordinates.

Given ω and $f_x = f_y = f_z = 0$, the equations (3.32) are linear and for given Δt, the time-invariant closed form solution of these differential equations can be written in the matrix form as:

$$\begin{bmatrix} \mathbf{r}_{k+1} \\ \mathbf{v}_{k+1} \end{bmatrix} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \begin{bmatrix} \mathbf{r}_k \\ \mathbf{v}_k \end{bmatrix}$$

Note that $\Phi_{rr}, \Phi_{rv}, \Phi_{vr}$ and Φ_{vv} are all invertible. Assuming an impulse control (as change in velocity) $\Delta\mathbf{v}$, 2 sequential impulses $\Delta\mathbf{v}_1$ and $\Delta\mathbf{v}_2$ are required to transition from state i to state j as:

$$\begin{bmatrix} \mathbf{r}_{j,i} \\ \mathbf{v}_{j,i} \end{bmatrix} = \Phi \begin{bmatrix} \mathbf{r}_i \\ \mathbf{v}_i \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta\mathbf{v}_1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{r}_j \\ \mathbf{v}_j \end{bmatrix} = \Phi \begin{bmatrix} \mathbf{r}_{j,i} \\ \mathbf{v}_{j,i} \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta\mathbf{v}_2 \end{bmatrix}$$

Moreover, these impulse control signals can be uniquely determined with the following relation,

$$\begin{bmatrix} \Delta\mathbf{v}_1 \\ \Delta\mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} \Phi_{rv} & \mathbf{0} \\ \Phi_{vv} & \mathbf{I} \end{bmatrix}^{-1} \left(\Phi^{-1} \begin{bmatrix} \mathbf{r}_j \\ \mathbf{v}_j \end{bmatrix} - \Phi \begin{bmatrix} \mathbf{r}_i \\ \mathbf{v}_i \end{bmatrix} \right). \tag{3.35}$$

The idea is to find the next SR coordinate using the density commands and the control law (3.23), then map this coordinate to the LVLH frame and calculate the necessary velocity to be in the invariant ellipse. Finally use this successor LVLH state information to calculate the required control signals and perform the maneuver. The algorithm can be summarized in the following steps:

For all spacecraft $i = 1, \ldots, N$,

1. At time t, map the current LVLH location $[x_i(t), y_i(t)]^\mathsf{T}$ to SR coordinate $[\bar{x}_i(t), \bar{y}_i(t)]^\mathsf{T}$ using (3.33), and broadcast this location to other spacecraft,

2. Estimate the current density in SR coordinates using (3.11) with the information received from other spacecraft,

3. Invoke the density based controller (3.23) and calculate the next SR coordinate $[\bar{x}_i(t + \Delta t), \bar{y}_i(t + \Delta t)]^\mathsf{T}$,

4. Map the next SR coordinate back to LVLH using,

$$\mathbf{r}_j = \begin{bmatrix} x_i(t+\Delta t) \\ y_i(t+\Delta t) - y_o \end{bmatrix} = SR\big(\omega(t+\Delta t)\big) \begin{bmatrix} \bar{x}_i(t+\Delta t) \\ \bar{y}_i(t+\Delta t) \end{bmatrix}$$

5. Calculate the velocity component $\mathbf{v}_j$ of the next state in LVLH using (3.34),

6. Compute the required 2 sequential impulse controls, i.e., $\Delta\mathbf{v}_1$ and $\Delta\mathbf{v}_2$ to achieve $[\mathbf{r}_j, \mathbf{v}_j]^\mathsf{T}$ using (3.35), and apply the control inputs with $\Delta t/2$ timesteps.

7. $t \leftarrow t + \Delta t$, and go back to Step 1.

Following this procedure, the swarm of spacecraft will converge to the desired density distribution in SR coordinates. The corresponding distribution in LVLH frame will be a stretched and constantly rotating version of the desired density with the orbit frequency of ω.

Orbital Simulation: Consider an OTM swarm of $N = 200$ spacecraft randomly deployed to a circular orbit at 500km altitude. Upon deployment, the swarm is commanded to a star density distribution, where the local density measurements are performed via Gaussian Kernel function with interpolation length $h = 20$. The controller gain is chosen as $\alpha = 2$. Timestep for each control signal is taken as 60 seconds, hence each density feedback is given in $\Delta t = 120$ seconds.

As it can be seen from Figure 3.7, the swarm converges to the desired density profile of star form. Also, the desired density distribution rotates in the orbital frame with the orbital frequency $\omega = 0.00110678$ rad/s which gives the orbital period $T = 5677$ s. The Figure 3.8 shows the stream of control signals in time. As the swarm approaches to the desired density profile, the control signals, i.e., changes in velocities dissipates.

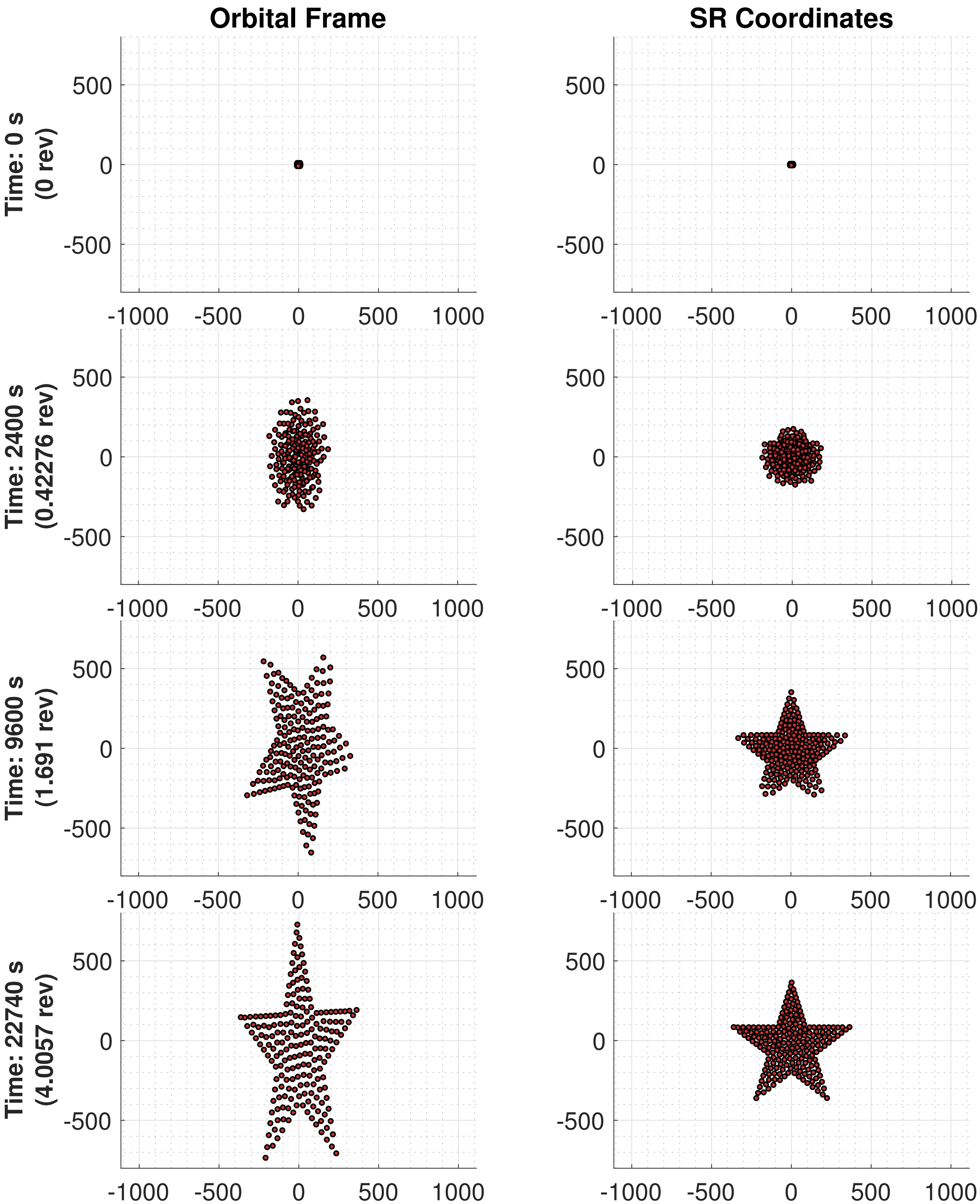

Figure 3.7: The evolution of the swarm through time illustrated in both LVLH frame and Scaled-Rotated coordinates.

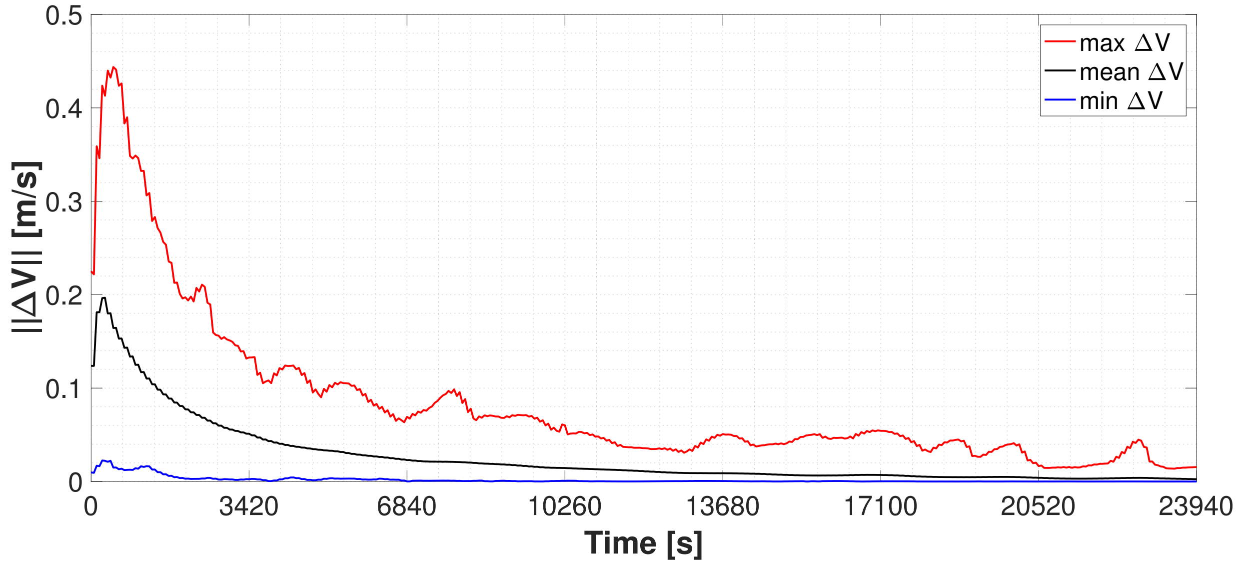

Figure 3.8: Control signals in time with maximum magnitude over the swarm(red), mean(black), and min(blue).

3.5.2 Pattern Generation via Drones

This demonstration aims to illustrate the velocity field based density control law (3.23) in a laboratory setting using 10 number of Crazyflie 2.0 quadrotors. Agents are commanded to generate patterns, specifically to create the letters A, C and L sequentially. For the control law, the *diffusion constant* α is chosen as 5. The density estimations are performed with 2-dimensional Gaussian kernel where the *interpolation length* h is chosen as 40 cm. Figure 3.9 provides the raw desired density commands that are read from images of the letters, and smooth versions of the raw input. Smooth density distributions are obtained via the spatial smoothing technique given in Section 3.4.1 using the same Gaussian kernel function that is used to estimate the density of agents.

Desired density commands are transformed to position commands using the heat equation based density feedback control law, and position commands are tracked by on-board PID controllers of the Crazyflie 2.0 quadrotors. Position feedback has been achieved with OptiTrack motion capture system, hence each quadrotor is equipped with unique 3D config-

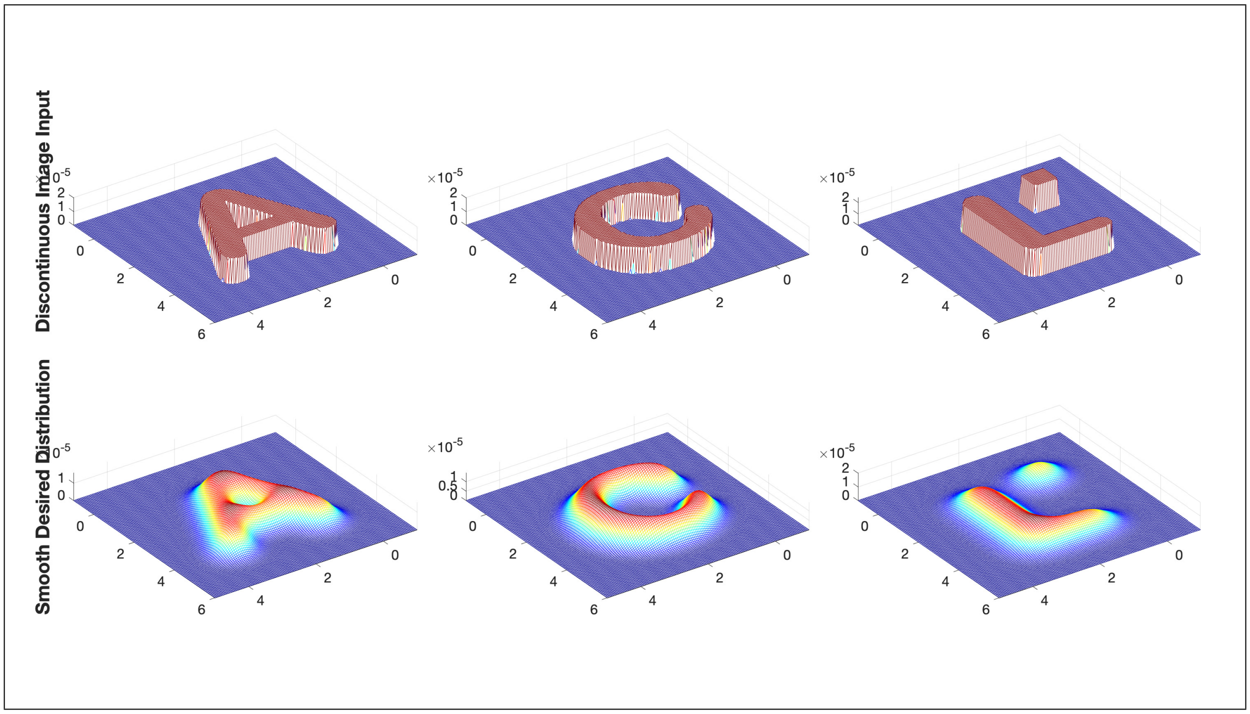

Figure 3.9: Raw Desired Density Commands read from images of letters A, C, L and the smooth counterparts for the raw input commands

uration of 4 reflective markers. It is important to note that the letter L has a less area given the letter widths are taken equal. Therefore, in order to preserve aspect ratio of the letter L and avoid agents getting too close to each other, some portion of the probability mass has been sent to top right corner in the frame of the letter L. Figure 3.11, presents the density estimations performed during A, C, L command sequence. Even though the maneuvers are performed in 3-dimensional space given by $6 \times 7 \times 3$ meters, the density estimations have been performed in $x - y$ space since the motion has been designed in 2-dimensional planar space with 30° inclination.

Figure 3.10 presents the snapshots from the laboratory demonstration. Full video for the generation of A, C, L sequence can be accessed via the link given in the caption. It is important to note that, mild perturbations have been observed during the experiment due to the disturbances and insufficiently tuned PID controllers after addition of 4 markers on

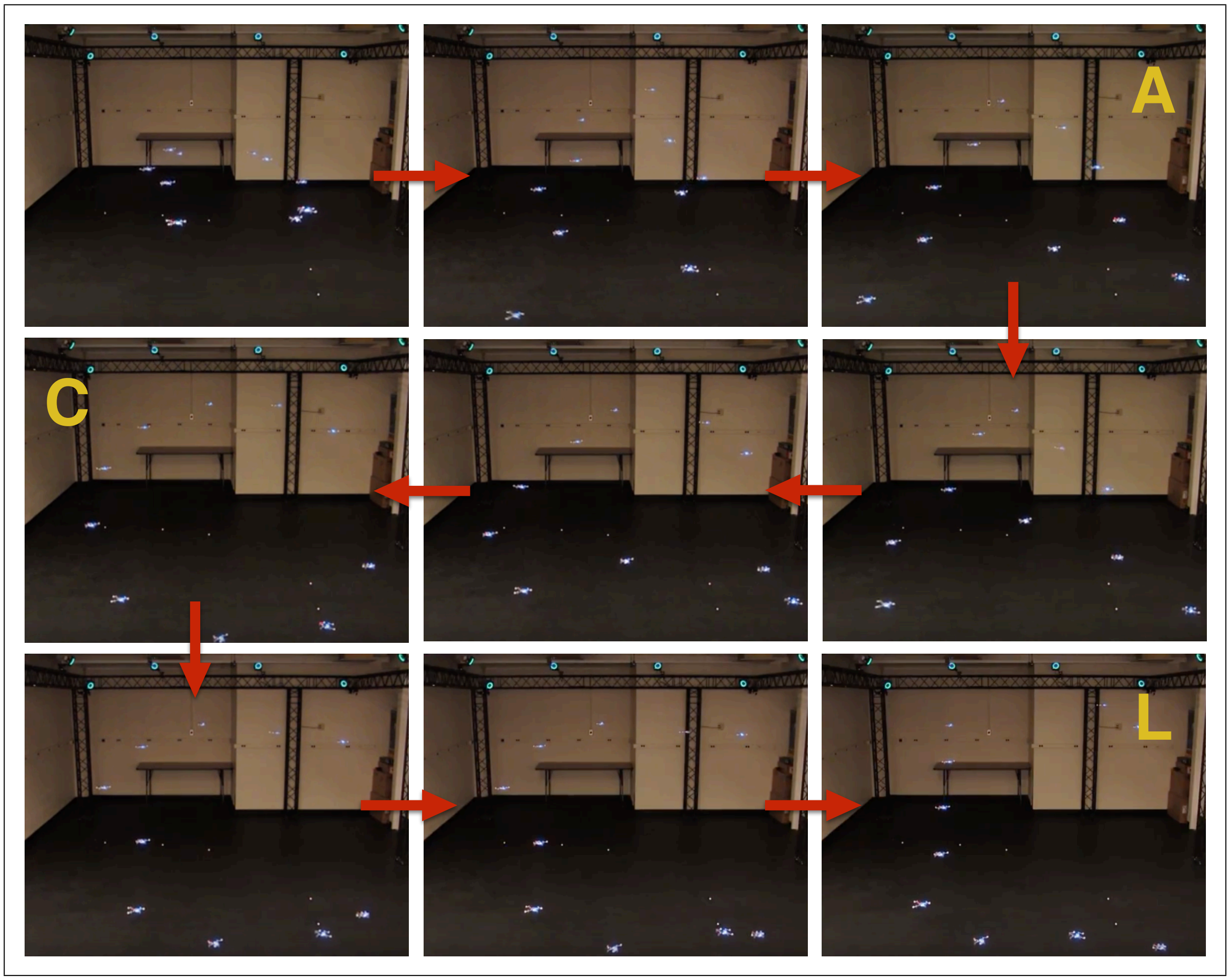

Figure 3.10: Deterministic Pattern Generation with 10 agents to form letter A, C and L respectively. For full video: `https://youtu.be/_Dxf_wx1184`

each quadrotor. Nevertheless agents settled in the desired *A, C, L* patterns sequentially.

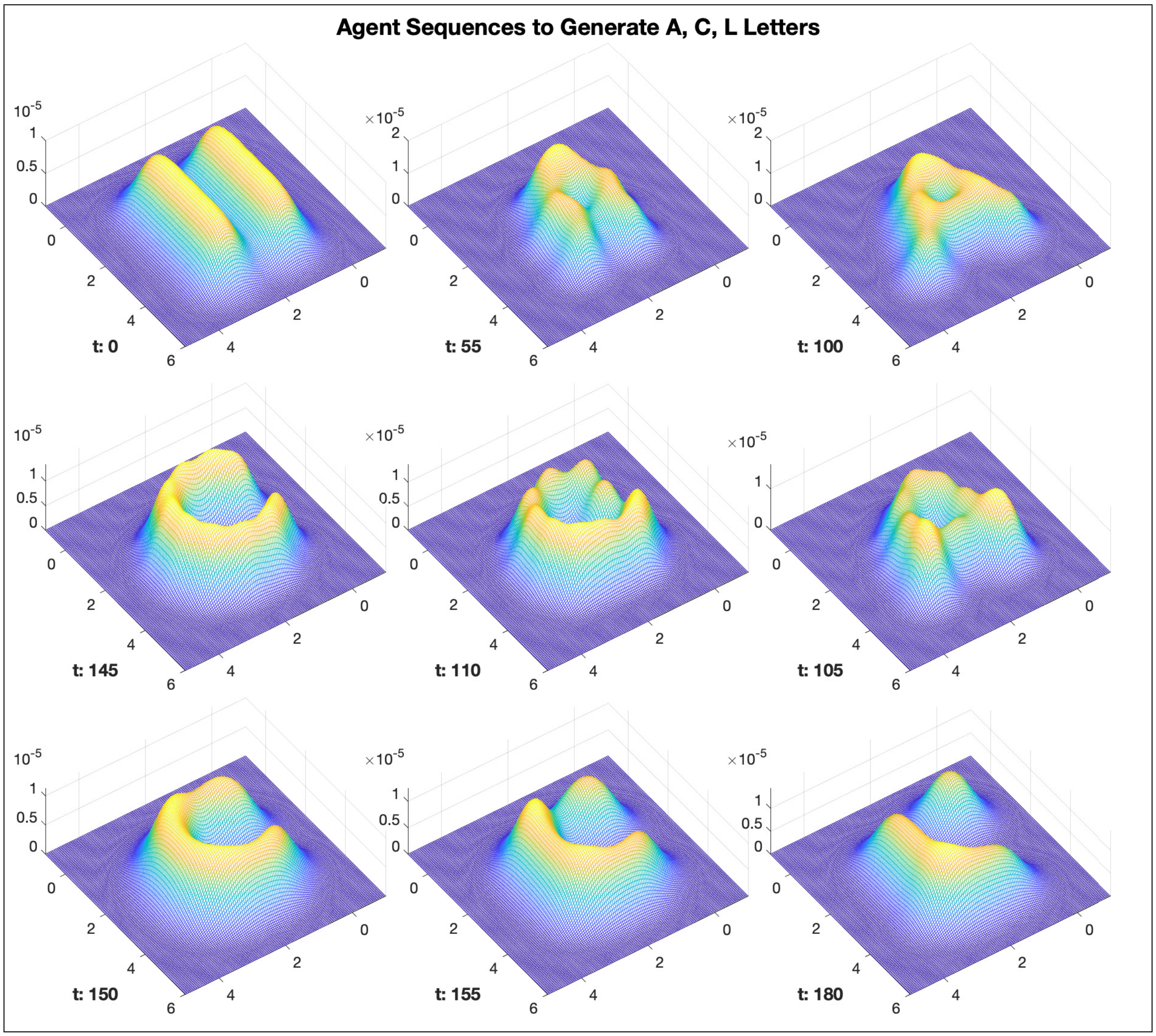

Figure 3.11: Kernel Density Estimations using Gaussian Kernels during the *A, C, L* sequence over the domain that 10 Crazyflie quadrotors perform their maneuvers.

Chapter 4

COLLISION AVOIDANCE ANALYSIS

In this chapter, collision avoidance guarantees are provided for density feedback control law with a certain class of kernel functions used in local density estimations. The lower bound on the minimum relative distance will be investigated using the dynamics of inter-agent distances. Furthermore, some observations regarding the relation between minimum collision radius and smoothing parameter h in density estimation (3.10) will be provided along with analytical expressions for determining the smoothing parameter to induce desired collision radius. Before the collision avoidance analysis, the following assumption will be employed throughout this research.

Assumption 2 *For any given $h > 0$, a differentiable kernel function $K_h(\mathbf{x})$ has Lipschitz continuous gradient $\nabla K_h(\mathbf{x})$ with Lipschitz constant of L_h, i.e.,*

$$\left\|\nabla K_h(\mathbf{x}) - \nabla K_h(\boldsymbol{y})\right\| \leq L_h \left\|\mathbf{x} - \boldsymbol{y}\right\|, \quad \forall \mathbf{x}, \boldsymbol{y} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$$

such that

$$\lim_{h \to 0} L_h = \infty.$$

Within the context of this work, Assumption 2 is a mild one in the sense that all the implementations and analyses of the proposed method are carried out with kernel functions that comply with this condition:

$$K_h(x) \;=\; \frac{1}{2h}\frac{1}{\Big(1+\big(|x|/h\big)\Big)^2}, \quad x \in \mathbb{R} \qquad \Rightarrow \qquad L_h = \frac{3}{h^3} \tag{4.1}$$

$$K_h(\mathbf{x}) \;=\; \frac{1}{C_d h^d}\exp\left(-\frac{\|\mathbf{x}\|}{h}\right), \quad \mathbf{x} \in \mathbb{R}^d \qquad \Rightarrow \qquad L_h = \frac{1}{C_d h^{d+2}}, \quad d \geq 1 \tag{4.2}$$

where $C_d = \{2, 2\pi, 8\pi, 12\pi^2, \dots\}$ for $d = 1, 2, \dots$. Also note that (4.1) is only a 1-dimensional kernel function, because, even if $|x|$ is replaced with $\|\mathbf{x}\|$, the kernel function does not satisfy the Condition (3.12b) for density estimation.

4.1 Background

Before the density feedback control is outlined and its collision avoidance guarantees are presented, the definitions and theorems used throughout the chapter are introduced. The chapter begins with the definition of *initial value problem* (also referred to as the *Cauchy problem*):

Definition 7 (Initial Value Problem) *Cauchy problem is a differential equation* $\dot{\mathbf{x}}(t) = f\big(t, \mathbf{x}(t)\big)$ *with* $f : \Omega \subset \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ *where* Ω *is an open set of* $\mathbb{R} \times \mathbb{R}^d$*, along with an initial condition* $\big(t_0, \mathbf{x}_0\big) \in \Omega$*.*

In order for an initial value problem to be *well-posed* [66], i.e., (i) a solution *exists*, (ii) the solution is *unique*, and (iii) the solution depends continuously on the initial conditions, one needs the following assumption [104].

Assumption 3 $f : \Omega \subset \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ *is continuous and satisfies the following locally Lipschitz condition: for each* $\big(t, \mathbf{x}\big) \in \Omega$*, there exist a neighborhood* $\mathcal{O}$ *of* $\big(t, \mathbf{x}\big)$ *in* Ω *and a constant* $L > 0$ *such that,*

$$\big\|f(s, y) - f(s, w)\big\| \leq L\big\|y - w\big\|, \quad \forall (s, y), (s, w) \in \mathcal{O}.$$

By Rademacher Theorem [97], any function that satisfies Assumption 3 is differentiable almost everywhere in Ω. The following result follows from Rademacher Theorem.

Proposition 2 *If $f : \Omega \to \mathbb{R}^d$ is differentiable and globally Lipschitz on Ω, then $\|\nabla f\|_{L^\infty(\Omega)} \leq L$, where L is the Lipschitz constant of f over Ω.*

We next provide the notion of *positive invariance* which is also often referred to as *flow invariance* [18].

Definition 8 *Let D be a nonempty subset of an open set $\mathcal{X} \subseteq \mathbb{R}^d$, given the differential equation,*

$$\dot{\mathbf{x}}(t) = f\big(\mathbf{x}(t)\big), \qquad t \geq 0, \tag{4.3}$$

with $f : \mathcal{X} \to \mathbb{R}^d$, the set D is said to be a flow-invariant set *with respect to (4.3) if every solution $\mathbf{x} : [0, T) \to \mathcal{X}$ of (4.3) with $\mathbf{x}(0) \in D$, verifies $\mathbf{x}(t) \in D$ for all $t \in [0, T)$.*

Flow invariance (or positive invariance) ensures that every solution of (4.3) starting from D remains in the set D, given that it exists. Before getting to the main theorem, we will provide the notion of *distance from a set* as follows:

Definition 9 *Given a set $D \subset \mathbb{R}^d$ and a point $y \in \mathbb{R}^d$, the distance between D and y is defined as, $\mathrm{dist}(y, D) = \inf_{w \in D} \|y - w\|_*$, where $\|\cdot\|_*$ is any relevant norm.*

Definition 10 *The nonempty set D is said to be* closed *in $\mathcal{X}$ if $D = \overline{D} \cap \mathcal{X}$, where $\overline{D}$ denotes the closure of D.*

The following theorem presents the *necessary and sufficient conditions* for positive invariance. It is an important result for the characterization of positively invariant sets for continuous time systems [95], which is also independently studied by [26, 21] and generalized by [92].

Theorem 6 (Weak Positive Invariance) *Given the differential equation (4.3), assume that for each initial condition $\mathbf{x}(0)$ in an open set $\mathcal{X}$, (4.3) admits a (not necessarily unique)* *solution defined for all $t \geq 0$. Let $D \subset \mathcal{X}$ be a closed set. Then, D is* weakly positively invariant *with respect to (4.3) if and only if:*

$$f(\mathbf{x}) \in \mathcal{T}_D(\mathbf{x}), \quad \forall \mathbf{x} \in D, \tag{4.4}$$

where $\mathcal{T}_D(\mathbf{x})$ is Bouligand's tangent cone [22] defined as,

$$\mathcal{T}_D(\mathbf{x}) = \left\{ z : \quad \liminf_{\tau \to 0} \frac{\text{dist}(\mathbf{x} + \tau z, D)}{\tau} = 0 \right\}. \tag{4.5}$$

Corollary 4 (Positive Invariance) *The positive invariance of D is equivalent to condition (4.4) if more strict assumption of* uniqueness of the solution *for every $\mathbf{x}(0) \in \mathcal{X}$ is satisfied under the assumptions/conditions of Theorem 6.*

As a consequence of Corollary 4, the notion of weak positive invariance turns into positive invariance when Assumption 3 is satisfied.

Remark 6 *It is important to note that Corollary 4 holds even if f is defined and locally Lipschitz only on D [105].*

As it is stated in [105], in case of Remark 6, Corollary 4 becomes a result of *existence* rather than a problem of *invariance*. In other words, if $D \subset \mathcal{X}$ is closed and f is globally Lipschitz on D, then the condition (4.4) guarantees the existence and uniqueness of solution $\mathbf{x}(t) : [0, +\infty) \to D$ to the differential equation (4.3) with $\mathbf{x}(0) = \mathbf{x}_0 \in D$ [92].

Note that the condition (4.4) is only meaningful when $\mathbf{x} \in \partial D$, since $\mathcal{T}_D(\mathbf{x})$ becomes $\mathbb{R}^d$ when $\mathbf{x} \in int\{D\}$. Under the assumption of uniqueness of the solution $\mathbf{x}(t)$, the condition (4.4) along with the definition of tangent cone (4.5) means that if the vector $\dot{\mathbf{x}} = f(\mathbf{x})$ *points inside or it is tangent to* D for all $\mathbf{x} \in \partial D$, then the trajectory $\mathbf{x}(t)$ remains in D for all t [18]. Now consider the case where the set D can be defined by a finite set of inequalities of the form:

$$D = \{\mathbf{x} : \; g_k(\mathbf{x}) \leq 0, \quad k = 1, 2, \ldots, c\} \tag{4.6}$$

where $g_k(\mathbf{x}) \in C^1(\mathcal{X}, \mathbb{R})\ \forall k$. Consequently, the interior of the set is given by,

$$int\{D\} = \{\mathbf{x}: \ g_k(\mathbf{x}) < 0, \quad k = 1, 2, \ldots, c\}.$$

Defining the set of active constraints as

$$Act(\mathbf{x}) = \{i: \ g_i(\mathbf{x}) = 0\},$$

simplifies the tangent cone (4.5) for $\mathbf{x} \in \partial D$ as follows [18]:

$$\mathcal{T}_D(\mathbf{x}) = \left\{z: \ \nabla g_i(\mathbf{x})^\mathsf{T} z \leq 0, \quad \forall i \in Act(\mathbf{x})\right\}. \tag{4.7}$$

4.2 Problem Formulation

The foundation of the collision avoidance analysis is to find a *lower bound* R_c on the minimum inter-agent distances among all agents at all times, i.e.,

$$\|\mathbf{x}_i(t) - \mathbf{x}_j(t)\|_2 \geq R_c, \quad \forall (i,j) \quad \text{and} \quad \forall t. \tag{4.8}$$

Let the constrained set of inter-agent distances be,

$$\mathcal{V}(R_c) \triangleq \{\mathbf{X} \in \mathbb{R}^{Nd} \mid \|\mathbf{x}_i - \mathbf{x}_j\|_2 \geq R_c \ \ \forall (i,j)\}, \tag{4.9}$$

where $\mathbf{X} = [\mathbf{x}_1^\mathsf{T}, \ldots, \mathbf{x}_N^\mathsf{T}]^\mathsf{T}$. The objective can be formulated as to find $R_c > 0$, such that $\mathcal{V}(R_c)$ is a *positive/flow invariant set* with respect to the dynamics that govern the motion of $\mathbf{x}_i(t)\ \forall i$. Defining $\mathbf{r}_{ij}(t) = \mathbf{x}_i(t) - \mathbf{x}_j(t)$, and dropping the time argument for notational convenience, the dynamics for inter-agent vectors can be stated as follows:

$$\dot{\mathbf{r}}_{ij} = f_{r_{ij}}(\mathbf{X}) = f_i(\mathbf{X}) - f_j(\mathbf{X}), \quad \forall (i,j), \tag{4.10}$$

where the individual agent dynamics are defined as,

$$\dot{\mathbf{x}}_i(t) = -\alpha\,\nabla\big(\rho(\mathbf{x}_i) - \rho_D(\mathbf{x}_i)\big). \tag{4.11}$$

Given that the dynamics (4.11) satisfies Assumption 3 then (4.10) also satisfies Assumption 3. Now observe that the set $\mathcal{V}(R_c)$ is written in terms of $N(N-1)/2$ number of inequalities, and it is a closed set by Definition 10. Therefore it fits the set representation (4.6) with $g_{ij}(\mathbf{r}_{ij}) = R_c - \|\mathbf{r}_{ij}\|_2$. Using Corollary 4 and the tangent cone definition (4.7) along with the assumption $R_c \leq \|\mathbf{r}_{ij}(0)\|_2\ \forall(i,j)$, the equivalent condition for positive invariance of $\mathcal{V}(R_c)$ can be written as follows,

$$f_{r_{ij}}(\mathbf{X}) \in \left\{z:\ \nabla_{\mathbf{r}_{ij}} g_{ij}(\mathbf{r}_{ij})^{\mathsf{T}} z \leq 0\right\},\quad \forall(i,j) \in Act(\mathbf{X}) \tag{4.12}$$

where $Act(\mathbf{X})$ is defined as,

$$Act(\mathbf{X}) = \{(i,j):\ g_{ij}(\mathbf{x}_i - \mathbf{x}_j) = 0\}.$$

Further simplification of

$$\nabla_{\mathbf{r}_{ij}} g_{ij}(\mathbf{r}_{ij})^{\mathsf{T}}\dot{\mathbf{r}}_{ij} \;=\; \nabla_{\mathbf{r}_{ij}}\big(R_c - \|\mathbf{r}_{ij}\|_2\big)^{\mathsf{T}}\dot{\mathbf{r}}_{ij} \;=\; -\frac{\mathbf{r}_{ij}^{\mathsf{T}}}{\|\mathbf{r}_{ij}\|_2}\dot{\mathbf{r}}_{ij} \;\leq\; 0,$$

yields the following form,

$$\mathbf{r}_{ij}^{\mathsf{T}}\dot{\mathbf{r}}_{ij} \;\geq\; 0, \qquad \forall(i,j) \in Act(\mathbf{X}) \tag{4.13}$$

Defining,

$$\eta_{ij}(t) \triangleq \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\big(\mathbf{r}_{ij}^{\mathsf{T}}\ \mathbf{r}_{ij}\big) = (\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j)^{\mathsf{T}}(\mathbf{x}_i - \mathbf{x}_j),$$

the positive invariance condition can be stated as

$$\|\mathbf{r}_{ij}(t)\|_2 = R_c \quad \Rightarrow \quad \eta_{ij}(t) \geq 0 \quad \forall (i,j). \tag{4.14}$$

The condition (4.14) states that, whenever any inter-agent distance drops down to lower bound R_c, the time derivative of the distance function acts to preserve or increase the distance. In order to show the guarantee for collision avoidance, i.e, existence of $R_c > 0$ under the dynamics (4.11), *(i)* first the *worst case swarm configuration* will be defined and identified, *(ii)* then the satisfaction of condition (4.14) under the worst case swarm configuration will be verified.

4.2.1 Worst Case Swarm Configuration

Without loss of generality, consider a case where $\exists (i,j)$ such that $\|\mathbf{r}_{ij}(0)\|_2 \;= R_c$. For a swarm of N agents, the velocities for i^{th} and j^{th} agents can be written as follows:

$$\begin{aligned} \dot{\mathbf{x}}_i &= -\alpha \left(\frac{1}{N} \sum_{l \neq i}^{N} \nabla K_h\big(\mathbf{r}_{il}\big) - \nabla \rho_D\big(\mathbf{x}_i\big) \right), \\ \dot{\mathbf{x}}_j &= -\alpha \left(\frac{1}{N} \sum_{l \neq j}^{N} \nabla K_h\big(\mathbf{r}_{jl}\big) - \nabla \rho_D\big(\mathbf{x}_j\big) \right). \end{aligned}$$

Assuming that the kernel function is even, i.e., $K_h\big(\mathbf{r}\big) = K_h\big(-\mathbf{r}\big)$, hence, $-\nabla K_h\big(\mathbf{r}\big) = \nabla K_h\big(-\mathbf{r}\big)$, the definition of η_{ij} can be stated as,

$$\begin{aligned} \eta_{ij} &= (\mathbf{x}_i - \mathbf{x}_j)^{\mathsf{T}} (\dot{\mathbf{x}}_i - \dot{\mathbf{x}}_j) \\ &= -\alpha \; \mathbf{r}_{ij}^{\mathsf{T}} \left(\frac{1}{N} \sum_{l \neq i}^{N} \nabla K_h\big(\mathbf{r}_{il}\big) - \frac{1}{N} \sum_{l \neq j}^{N} \nabla K_h\big(\mathbf{r}_{jl}\big) - \Big(\nabla \rho_D(\mathbf{x}_i) - \nabla \rho_D(\mathbf{x}_j) \Big) \right), \end{aligned}$$

where self (i,j) term can be seperated from the cross terms as follows,

$$\eta_{ij} = -\alpha \Bigg(\frac{2}{N}\mathbf{r}_{ij}^{\mathsf{T}} \nabla K_h\big(\mathbf{r}_{ij}\big) \tag{4.15a}$$

$$+ \frac{1}{N}\mathbf{r}_{ij}^{\mathsf{T}} \sum_{l \neq i,j} \Big(\nabla K_h\big(\mathbf{r}_{il}\big) - \nabla K_h\big(\mathbf{r}_{jl}\big)\Big) \tag{4.15b}$$

$$- \mathbf{r}_{ij}^{\mathsf{T}} \Big(\nabla \rho_D(\mathbf{x}_i) - \nabla \rho_D(\mathbf{x}_j)\Big)\Bigg) \geq 0. \tag{4.15c}$$

Notice that, in the definition of η_{ij}, the term (4.15a) is a function of relative location $\mathbf{r}_{ij}$, the term (4.15b) is defined by the relative configuration of other agents' locations $\{\mathbf{r}_{il}, \mathbf{r}_{jl}\}$ $\forall l \neq \{i,j\}$, and the term (4.15c) is determined by the desired density $\rho_D(\mathbf{x})$ at the location of i^{th} and j^{th} agents. Now, the worst possible configuration of agents in a swarm can be stated in terms of η_{ij} as follows.

Definition 11 (Worst Case Swarm Configuration) *Given a relative location $\mathbf{r}_{ij}$ and desired density $\rho_D(\mathbf{x})$, worst case swarm configuration for the pair of agents (i,j) is the configuration of all agents that minimizes η_{ij}, i.e.,*

$$\begin{aligned} \bar{\eta}_{ij} = & -\frac{2}{N}\mathbf{r}_{ij}^{T} \nabla K_h\big(\mathbf{r}_{ij}\big) \\ & - \frac{1}{N} \max_{\mathbf{x}_l} \left\{ \mathbf{r}_{ij}^{T} \sum_{l \neq i,j} \Big(\nabla K_h\big(\mathbf{r}_{il}\big) - \nabla K_h\big(\mathbf{r}_{jl}\big)\Big) \right\} \\ & + \min_{\mathbf{x}_i, \mathbf{x}_j} \left\{ \mathbf{r}_{ij}^{T} \Big(\nabla \rho_D\big(\mathbf{x}_i\big) - \nabla \rho_D\big(\mathbf{x}_j\big)\Big) \right\}. \end{aligned} \tag{4.16}$$

The argument pertaining to *worst case swarm configuration* is that, if the $\bar{\eta}_{ij} \geq 0$ is satisfied, then $\eta_{ij} \geq 0$ will be satisfied for any agent configuration in a swarm at any time. The identification of worst case swarm configuration can be performed in 2 consecutive steps:

Step 1: Given $\mathbf{r}_{ij}$ and $\rho_D(\mathbf{x})$, the worst case swarm configuration of $\mathbf{x}_i$ and $\mathbf{x}_j$ can be calculated by performing the following optimization problem:

$$\min_{\mathbf{x}_i,\mathbf{x}_j} \quad \mathbf{r}_{ij}^{\mathsf{T}}\Big(\nabla\rho_D\big(\mathbf{x}_i\big) - \nabla\rho_D\big(\mathbf{x}_j\big)\Big)$$

$$\text{s.t.} \qquad \|\mathbf{r}_{ij}\|_2 = R_c$$

Remark 7 *In the case of uniform desired density, i.e.,* $\nabla\rho_D(\mathbf{x}) = 0\ \forall\mathbf{x}$*, the values of* $\mathbf{x}_i$ *and* $\mathbf{x}_j$ *can be chosen arbitrarily with the constraint of* $\|\mathbf{r}_{ij}\|_2 = R_c$*.*

Step 2: Once the information of $\mathbf{x}_i$ and $\mathbf{x}_j$ is obtained, the identification of the worst case swarm configuration for the rest of the agents can be obtained with the following optimization problem:

Optimization Problem 3 *Worst Case Swarm Configuration*

$$\max_{\mathbf{X}\setminus\{\mathbf{x}_i,\mathbf{x}_j\}} \quad \mathbf{r}_{ij}^{T}\left(\sum_{l\neq i,j}\nabla K_h\big(\mathbf{r}_{il}\big) - \sum_{l\neq i,j}\nabla K_h\big(\mathbf{r}_{jl}\big)\right)$$

$$\text{s.t.} \qquad \|\mathbf{r}_{ij}\|_2 \leq \|\mathbf{r}_{kl}\|_2, \qquad \forall(k,l)$$

Given that $\|\mathbf{r}_{ij}(0)\|_2 = R_c$, the set of constraints in Problem 3 ensure that $R_c \leq \|\mathbf{r}_{kl}(0)\|_2\ \forall(k,l)$. Solving for Problem 3 yields $\mathbf{x}_l(0)$ where $l \neq \{i,j\}$, and it only provides the worst case swarm configuration for the agents i and j. In other words, the solution yields the worst case swarm configuration for arbitrary pair of agents. In order to show that $\mathcal{V}(R_c)$ is a positive invariant set even for worst case swarm configuration one needs the following result.

Lemma 6 *Given* $\mathbf{x}_i(0)$ *and* $\mathbf{x}_j(0)$ *with* $\|\mathbf{r}_{ij}(0)\|_2 = R_c$*, and* $\mathbf{x}_l(0)$ *where* $l \neq \{i,j\}$ *as a solution of Problem 3, define* $\mathcal{V}_{ij}(R_c) \triangleq \{\mathbf{X} \in \mathbb{R}^{Nd} \mid \|\mathbf{x}_i - \mathbf{x}_j\|_2 \geq R_c\}$*. If* $\mathcal{V}_{ij}(R_c)$ *is a positive invariant set under (4.11) then* $\mathcal{V}(R_c)$ *is also positive invariant.*

Proof: The solution of Problem 3 yields initial conditions for agents where $R_c \leq \|\mathbf{r}_{kl}(0)\|_2,\ \forall(k,l)$, hence, $\mathbf{X}(0) \in \mathcal{V}(R_c)$. Now, it needs to be shown that, $\mathbf{X}(t) \in \mathcal{V}(R_c),\ \forall t >$

0. Since $\mathcal{V}_{ij}(R_c)$ is a positive invariant set under (4.11), by the definition of (4.16), the condition

$$\|\mathbf{r}_{ij}(t)\|_2 = R_c \quad \Rightarrow \quad \eta_{ij}(t) \geq \bar{\eta}_{ij}(0) \geq 0 \tag{4.17}$$

is satisfied for $\forall t \geq 0$. Now observe that (4.17) is also satisfied for $\mathbf{r}_{kl}(t),\ \forall(k,l),\ \forall t \geq 0$, because Problem 3 yields the worst possible configuration for an arbitrary pair in the swarm and there cannot be a pair in the swarm which is in even worse condition than the pair (i,j). In other words, $\bar{\eta}_{ij}(0)$ is the lower bound for any $\eta_{kl}(t)$, i.e.,

$$\eta_{kl}(t) \geq \bar{\eta}_{ij}(0) \geq 0, \qquad \forall(k,l), \text{ and } \forall t \geq 0.$$

Therefore, even in the worst case swarm configuration for (i,j), the condition (4.14) is satisfied for all pairs of agents, hence $\mathbf{X}(t) \in \mathcal{V}(R_c),\ \forall t \geq 0$. ■

4.2.2 KKT Conditions for Problem 3

In order to identify the worst case configuration of agents in a swarm, the *Karush-Kuhn-Tucker* (KKT) conditions of Problem 3 will be used. Define the Lagrangian as follows:

$$L(\mathbf{X},\mathbf{u}) \;=\; \mathbf{r}_{ij}^{\mathsf{T}}\left(\sum_{l\neq i,j}\nabla K_h\big(\mathbf{r}_{il}\big) - \sum_{l\neq i,j}\nabla K_h\big(\mathbf{r}_{jl}\big)\right) \;+\; \sum_{k,l\neq i,j}\left(u_{kl}\Big(\big\|\mathbf{r}_{ij}\big\|^2 - \big\|\mathbf{r}_{kl}\big\|^2\Big)\right),$$

where $\mathbf{u} \in \mathbb{R}^{\frac{N(N-1)}{2}-1}$ and it is the vector of dual variables for all inter-agent distance constraints except $\mathbf{r}_{ij}$. Then the KKT conditions are:

$$\nabla_{\mathbf{x}_l} L \;=\; \mathbf{J}_{\mathbf{x}_l}\Big[\nabla K_h\big(\mathbf{r}_{il}\big) - \nabla K_h\big(\mathbf{r}_{jl}\big)\Big]\;\mathbf{r}_{ij} \;+\; 2\left(\sum_{k<l}^{N} u_{kl}\big(\mathbf{r}_{kl}\big) + \sum_{k>l}^{N} u_{lk}\big(\mathbf{r}_{lk}\big)\right) \;=\; 0,$$

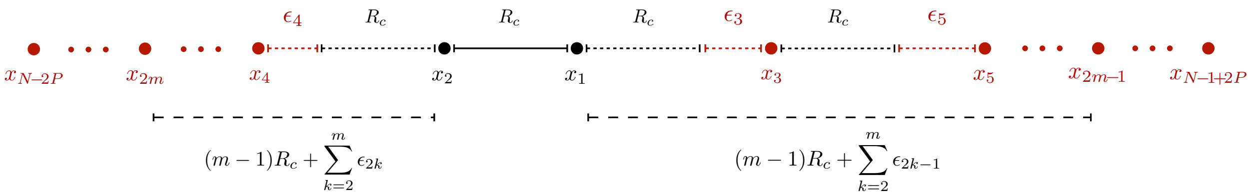

Figure 4.1: Parametrization of all 1-dimensional swarm configurations with respect to $N = 2M$, $P \in \{-M+1, -M+2, \ldots, M-1\}$, and $\epsilon_k \geq 0 \ \forall k$.

$$\left.\begin{aligned} u_{kl}\left(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{kl}\right\|^2\right) &= 0 \\ u_{kl} &\geq 0 \end{aligned}\right\} \qquad \forall k : k < l,$$

$$\left.\begin{aligned} u_{lk}\left(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{lk}\right\|^2\right) &= 0 \\ u_{lk} &\geq 0 \end{aligned}\right\} \qquad \forall k : l < k.$$

$\forall l \neq \{i, j\}$, where $\mathbf{J}_{\mathbf{x}_l}$ is the Jacobian operator with respect to $\mathbf{x}_l$. Solving for these equations for arbitrary number of dimensions is a challenging task. The solution for $d = 1$ case will be provided first, then a conservative generalization will be presented for arbitrary number of dimensions to prove collision avoidance property.

4.3 Analysis for 1-Dimensional Case

In the following analysis, only 1-dimensional problem is considered and Problem 3 is transformed into a simpler representation. Prior to doing so, the following assumption on the kernel function is employed to eliminate conservatism in the analysis.

Assumption 4 $K_h : \mathbb{R} \to [0, \infty)$ *is a symmetric continuous kernel function such that:*

$$\left.\frac{d^2}{dx^2} K_h(x)\right|_{x=x_1} < \left.\frac{d^2}{dx^2} K_h(x)\right|_{x=x_2} \quad \text{for all} \quad |x_1| > |x_2|$$

where $x_1, x_2 \neq 0$, i.e., $\frac{d^2}{dx^2} K_h(x) = \nabla^2 K_h(x)$ strictly decreases away from 0.

Assuming that $N = 2M$, is an even number, and without loss of generality $i = 1$ and $j = 2$, the objective of Problem 3 for all possible agent configurations can be stated around x_1 and x_2 as follows:

$$J_{N,P} = r_{12}\left[\sum_{m=2}^{M+P}\Big(\nabla K_h\big(r_{1,2m\text{–}1}\big) - \nabla K_h\big(r_{2,2m\text{–}1}\big)\Big) \;+\; \sum_{m=2}^{M-P}\Big(\nabla K_h\big(r_{1,2m}\big) - \nabla K_h\big(r_{2,2m}\big)\Big)\right] \tag{4.18}$$

where $r_{ij} = x_i - x_j$, $P \in \{–M+1, –M+2, \dots, M-1\}$, and,

$$\begin{aligned} r_{1,2m\text{–}1} &= -\left((m-1)R_c + \sum_{k=2}^{m} \epsilon_{2k-1}\right), \\ r_{2,2m\text{–}1} &= -\left(mR_c + \sum_{k=2}^{m} \epsilon_{2k-1}\right), \\ r_{1,2m} &= mR_c + \sum_{k=2}^{m} \epsilon_{2k}, \\ r_{2,2m} &= (m-1)R_c + \sum_{k=2}^{m} \epsilon_{2k}, \end{aligned} \tag{4.19}$$

where $\epsilon_k \in [0, \infty)\ \forall k$, as it can be seen from Figure 4.1. The introduction of perturbation terms $\epsilon_k \geq 0$ is performed in such a way that the resulting swarm configuration is guaranteed to satisfy the constraints of Problem 3. Following this parametrization,

1. it will be shown that for any value of $P \in \{–M+1, –M+2, \dots, M-1\}$, the maximum $J^*_{N,P}$ is attained when $\epsilon_k = 0, \quad \forall k \in \{3, \dots, N\}$,

2. then it will be shown that $P^* = \arg\max_P \ J^*_{N,P} = 0$.

Lemma 7 *For all $P \in \{–M+1, –M+2, \dots, M-1\}$, the objective $J_{N,P}$ attains its maximum when $\epsilon_k = 0\ \forall k$.*

Proof: Deriving the gradient of (4.18) yields:

$$\begin{aligned}\frac{\partial J_{N,P}}{\partial \epsilon_{2n-1}} &= r_{12}\sum_{m=n}^{M+P}\Bigg(-\nabla^2 K_h\Big(-(m-1)R_c-\sum_{k=2}^{m}\epsilon_{2k-1}\Big) + \nabla^2 K_h\Big(-mR_c-\sum_{k=2}^{m}\epsilon_{2k-1}\Big)\Bigg)\\ \frac{\partial J_{N,P}}{\partial \epsilon_{2n}} &= r_{12}\sum_{m=n}^{M-P}\Bigg(\nabla^2 K_h\Big(mR_c+\sum_{k=2}^{m}\epsilon_{2k}\Big) - \nabla^2 K_h\Big((m-1)R_c+\sum_{k=2}^{m}\epsilon_{2k}\Big)\Bigg).\end{aligned}$$

Since $K_h(x)$ is symmetric, $\nabla^2 K_h(-x) = \nabla^2 K_h(x)$. Given that $m \geq 2$, $R_c > 0$, and following Assumption 4, one can see that for all $\epsilon_k \geq 0$, all gradients are negative, i.e.,

$$\frac{\partial J_{N,P}}{\partial \epsilon_{2n-1}} < 0, \qquad \frac{\partial J_{N,P}}{\partial \epsilon_{2n}} < 0, \quad \forall n.$$

Since there are no saddle points in the half-closed interval of $[0,\infty)$, the maximum is achieved at the boundary, and since the gradient is negative for all $\epsilon_k \geq 0$ and all $P \in \{-M+1, -M+2, \ldots, M-1\}$, the maximum is achieved at $\epsilon_k = 0\ \forall k \in \{3,\ldots,N\}$. ■

Proposition 3 *For the class of $K_h : \mathbb{R} \to [0,\infty)$ that satisfies Assumption 4 the following holds for $x_1, x_2 \in \mathbb{R}_+$:*

$$\nabla K_h(x_1) + \nabla K_h(x_2) \leq 2\nabla K_h\big((x_1+x_2)/2\big)$$

Proof: Since $\nabla^2 K_h(x)$ is strictly decreasing on $\mathbb{R}_+$, $\nabla K_h(x)$ is a concave function on $\mathbb{R}_+$, hence the inequality follows. ■

Lemma 8 *For the class of $K_h : \mathbb{R} \to [0,\infty)$ that satisfies Assumption 4, the worst case swarm configuration in case of $d = 1$ is given as follows:*

$$x_{2m-1} = \left(m-\frac{1}{2}\right)R_c \qquad x_{2m} = -\left(m-\frac{1}{2}\right)R_c \tag{4.20}$$

for $m = 1,\ldots,M$, where $N = 2M$.

Proof: Using the objective definition (4.18), define

$$P^* = \arg\max_{P} \; J_{N,P}, \qquad \text{s.t. } \epsilon_k = 0 \;\; \forall k \in \{3,\ldots,N\},$$

it will be shown that $P^* = 0$. Now, observe the following:

$$\begin{aligned}
\Delta J &= J_{N,P} - J_{N,0} \\
&= r_{12}\Bigg[\sum_{m=M+1}^{M+P} \nabla K_h\big(r_{1,2m-1}\big) - \nabla K_h\big(r_{2,2m-1}\big) \; - \sum_{m=M-P+1}^{M}\Big(\nabla K_h\big(r_{1,2m}\big) - \nabla K_h\big(r_{2,2m}\big)\Big)\Bigg] \\
&= r_{12}\Bigg[\sum_{m=M+1}^{M+P} \nabla K_h\big(mR_c\big) - \nabla K_h\big((m-1)R_c\big) + \sum_{m=M-P+1}^{M}\Big(\nabla K_h\big((m-1)R_c\big) - \nabla K_h\big(mR_c\big)\Big)\Bigg],
\end{aligned}$$

cancelling terms in series yields the following,

$$\Delta J = r_{12}\big[\nabla K_h\big((M+P)R_c\big) + \nabla K_h\big((M-P)R_c\big) - 2\nabla K_h\big(MR_c\big)\big] \leq 0,$$

for any $P \in \{-M+1, -M+2, \ldots, M-1\}$. The last inequality is due to Proposition 3. Setting $x_1 = R_c/2$ and $x_2 = -R_c/2$, the equations (4.20) follow from (4.19). ■

Remark 8 *Note that the solution given by (4.20) is one of the equal objective value solutions, as changing the indices among agents would yield a different solution with the same objective value. The agent indices in Lemma 8 are chosen for convenience.*

Upon identification of the worst case swarm configuration, if one can find $R_c > 0$, such that the condition (4.14) is satisfied, then the collision avoidance guarantee will be established.

Corollary 5 *Given $K_h(x) : \mathbb{R} \to [0, \infty)$ that satisfies Assumption 4, and a desired density function with Lipschitz continuous gradient $\nabla \rho_D(x)$, the upper bound on the maximum feasible collision radius R_c for agents following the dynamics (4.11) is given by the following* *implicit inequality:*

$$\nabla K_h\big(MR_c\big) + MR_c L_{\rho_D} \leq 0, \tag{4.21}$$

where L_{ρ_D} is the Lipschitz constant for the desired density gradient $\nabla\rho_D(x)$.

Proof: The inequality is obtained using the result in Lemma 8 with Definition 11, and substituting (4.20) in (4.16) as follows:

$$\begin{aligned}
\bar{\eta}_{12} = & -\frac{2}{N}R_c\nabla K_h\big(R_c\big) \;-\; \frac{1}{N}R_c\Big(-2\nabla K_h\big(R_c\big) + 2\nabla K_h\big(MR_c\big)\Big) \\
& + \min_{\mathbf{x}_i,\mathbf{x}_j}\left\{\mathbf{r}_{ij}^{\mathsf{T}}\Big(\nabla\rho_D\big(\mathbf{x}_i\big) - \nabla\rho_D\big(\mathbf{x}_j\big)\Big)\right\} \\
= & -\frac{2}{N}R_c\nabla K_h\big(MR_c\big) - {R_c}^2 L_{\rho_D} \geq 0,
\end{aligned}$$

$$\Rightarrow -\nabla K_h\big(MR_c\big) \geq MR_c L_{\rho_D}. \qquad \blacksquare$$

The inequality (4.21) is kernel specific and can be solved for a preferred kernel function. The next result shows the existence of $R_c > 0$ for (4.1).

Theorem 7 *Given a swarm of agents $\mathcal{S}(t) = \{x_1(t), x_2(t), \ldots, x_N(t)\}$, $\quad x_i(t) \in \mathbb{R}\ \forall i$, where,*

(i) the motion of each is governed by (4.11), with

$$\min_{1\leq i\neq j\leq N}\big|r_{ij}(0)\big| \geq R_c,$$

(ii) agents perform kernel density estimation using (4.1),

(iii) the desired density commands have Lipschitz continuous gradient $\nabla\rho_D(x)$ with $L_{\rho_D} \geq 0$,

then $\exists R_c > 0$, such that

$$\min_{1\leq i\neq j\leq N}\big|r_{ij}(t)\big| \geq R_c \quad \forall t \geq 0.$$

Proof: Using the Corollary 5, and rewriting (4.21) for (4.1), one has

$$\frac{h}{(MR_c+h)^3} - MR_c L_{\rho_D} \geq 0, \tag{4.22}$$

where $N = 2M$. Define $\xi = MR_c/h$ and $\hat{L} = h^3 L_{\rho_D}$, and note that for any $h > 0$ and $R_c > 0$ one has $\xi > 0$ and $\hat{L} \geq 0$, then

$$\frac{1}{(\xi+1)^3} - \xi\hat{L} \geq 0 \quad \Rightarrow \quad 1 - \xi(\xi+1)^3\hat{L} \geq 0,$$

using inequality $(\xi+1)^4\hat{L} \geq \xi(\xi+1)^3\hat{L}$ yields,

$$\begin{aligned} 1 \geq (\xi+1)^4\hat{L} \quad &\Rightarrow \quad \sqrt[4]{\frac{1}{\hat{L}}} - 1 \geq \xi \\ &\Rightarrow \quad \frac{2h}{N}\left(\sqrt[4]{\frac{1}{h^3 L_{\rho_D}}} - 1\right) \geq R_c. \end{aligned}$$

The last inequality gives an upper bound for R_c and it follows that $\forall h \in \left(0, \sqrt[3]{\frac{1}{L_{\rho_D}}}\right)$ there exists $R_c > 0$. ■

Remark 9 *An intiutive observation from the collision avoidance guaranteeing interval of h values is that,*

$$\lim_{L_{\rho_D}\to 0} \max\ h = \infty, \qquad \text{and} \qquad \lim_{L_{\rho_D}\to\infty} \max\ h = 0,$$

where the first relation suggests that for uniform density case, i.e. $L_{\rho_D} = 0$*, any interpolation length h complies with positive invariance condition, hence starting from initial configuration* $\mathcal{S}(0) = \{x_1(0), x_2(0), \ldots, x_N(0)\}$*, the swarm will keep expanding and all inter-agent distances will keep increasing.*

Also it is important to note that, for (4.1) the interval $h \in (0, \sqrt[3]{1/L_{\rho_D}})$ is still conservative for any $L_{\rho_D} > 0$ due to the polynomial inequality used in the proof. Nevertheless, the theorem provides an existence result for $R_c > 0$ and guarantees collision avoidance property of the control law (4.11) with kernel function (4.1). If one wants to solve the exact interval for h, the inequality (4.22) needs to be solved for h numerically for given $N = 2M$, L_{ρ_D} and preferred R_c. Demonstration of this process will be presented in numerical examples.

Numerical Examples

Consider the kernel function (4.1) which satisfies (3.12), (3.16) and Assumption 4. First we will show how the inequality (4.22) can be used to choose h for a desired collision radius, with respect to the Lipschitz constant of the desired density gradient. Then, we will simulate (4.11) for arbitrary h values and show the collision radius predicted by (4.22) is satisfied.

Design Example: The goal of the demonstration is to show how to pick interpolation parameter h to induce the desired collision radius R_c. For that purpose, we have chosen three different Lipschitz constants $L \in \{0.0001, 0.01, 1.0, 10.0\}$ for desired density gradients $\nabla\rho_D(x)$, and three different numbers of agents $N \in \{10, 30, 100\}$. Figure 4.2 shows the R_c profiles with respect to h for all combinations of L and N. As it can be seen from the figure, as the Lipschitz constant for $\nabla\rho_D(x)$ increases, the collision radius that (4.11) provides decreases, i.e., higher variance in the desired density gradients causes a smaller collision radius that agents can have. Similarly, as the number agents increases to create the same density pattern, R_c decreases since the same pattern will be more dense in terms of the number of agents it has been formed with.

Statistical Simulation Example: In this example, we performed 50 simulations for each $h \in \{1, 2, 3, 4, 6, 8, 12\}$ values, using the following desired density function:

$$\rho_D(x) = \frac{1}{(\sigma\sqrt{\pi})} \exp\left(-\frac{x^2}{\sigma^2}\right),$$

where $\sigma = 35$. We predicted R_c using (4.22) and initiated all the simulations from randomly sampled initial agent locations that respects $\|\mathbf{x}_i(0) - \mathbf{x}_j(0)\|_2 \geq R_c$, $\forall(i, j)$. Minimum,

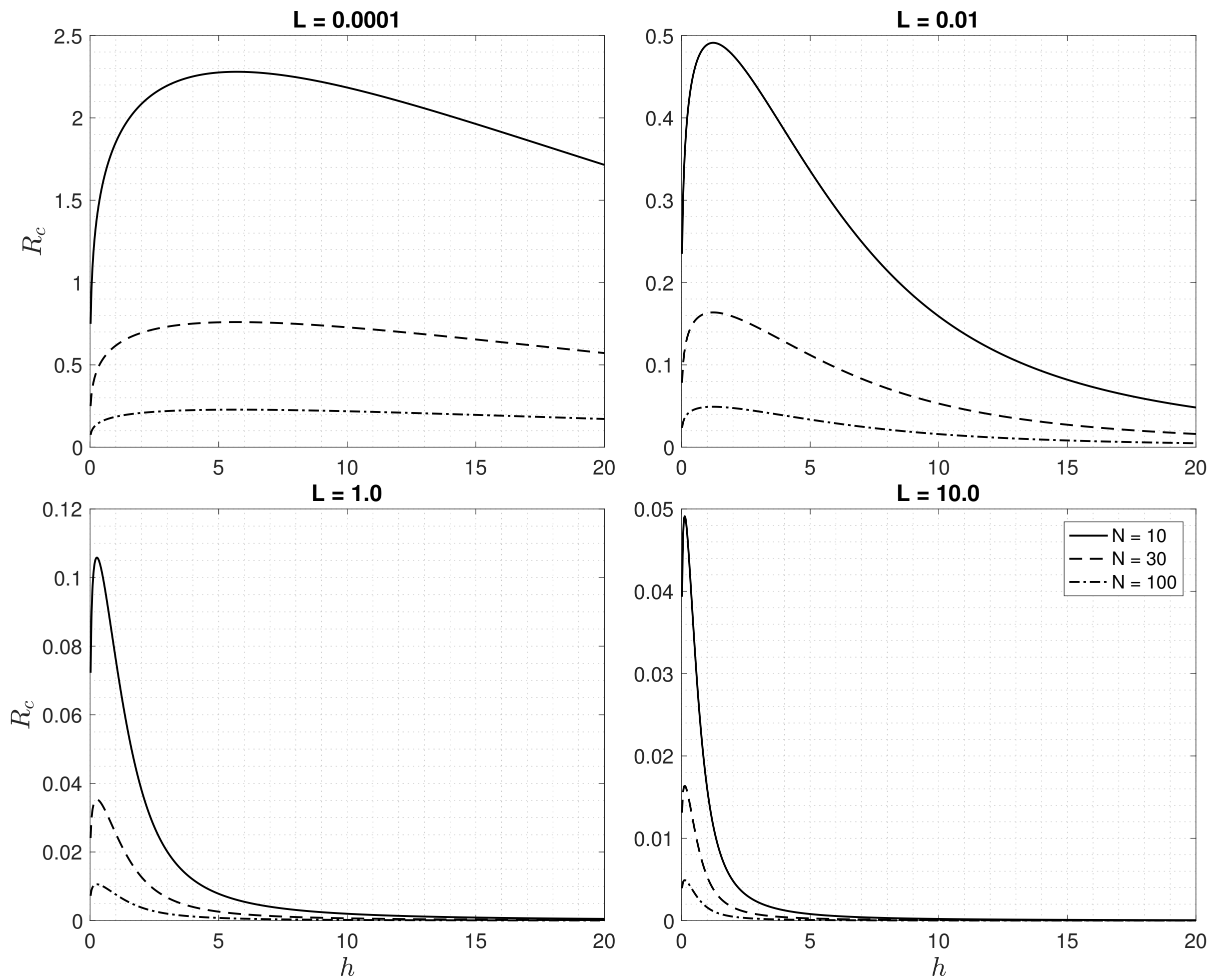

Figure 4.2: Collision radius R_c with respect to interpolation length h for $L \in \{0.0001, 0.01, 1.0, 10.0\}$ and $N \in \{10, 30, 100\}$ cases.

maximum, and mean values of the observed inter-agent distances are recorded and visualized in Figure 4.3, where the red dashed line represents the R_c values predicted by (4.22). As the figure illustrates, the minimum recorded inter-agent distances are always greater than theoretical bound given by (4.22). The figure on the right presents output of one of the simulations for $h = 8$ case. The initial condition (blue dots) propagated by (4.11) and the steady-state configuration (red dots) satisfied the desired density configuration.

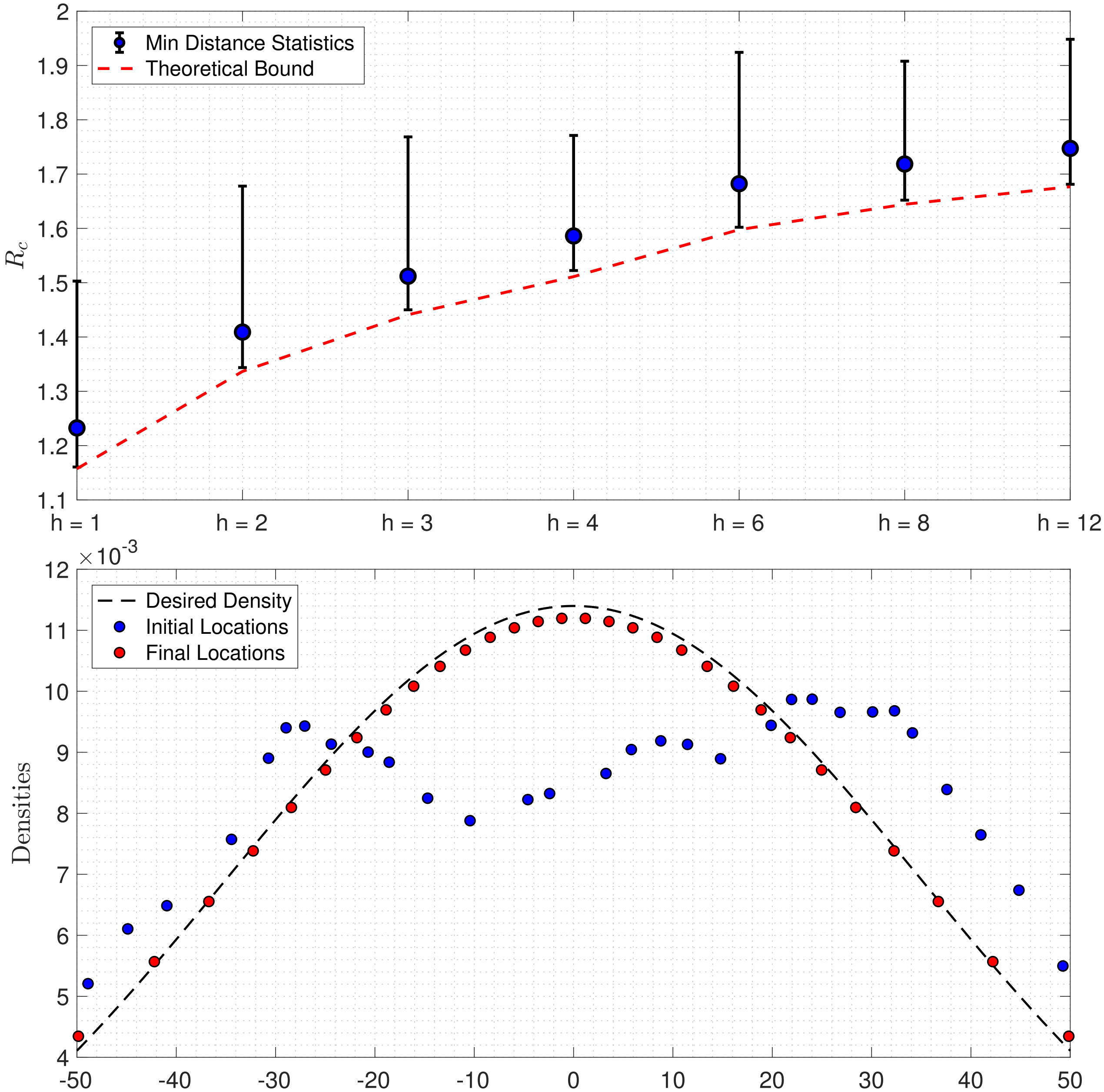

Figure 4.3: *Top*: Simulations with different h parameters. For each $h \in \{1, 2, 3, 4, 6, 8, 12\}$ 50 simulations with different initial conditions were run, and minimum observed inter-agent distance was recorded and compared with what (4.22) predicts. *Bottom*: Output of one of the simulations for $h = 8$ case. Blue dots are the initial, red dots are the steady-state agent locations of (4.11) with density values calculated at agent locations, and dashed line is $\rho_D(x)$.

4.4 Conservative Generalization to Higher Dimensions

Pursuing the solution of Problem 3, especially for number of dimensions where $d > 1$, is challenging. In order to make the problem more tractable, the following formulation will be employed where the constraints between agents other than i^{th} and j^{th} are relaxed as follows:

Optimization Problem 4 *Relaxed Worst Case Swarm Configuration*

$$\max_{\mathbf{X}\setminus\{\mathbf{x}_i,\mathbf{x}_j\}} \quad \mathbf{r}_{ij}^T \left(\sum_{l\neq i,j} \nabla K_h\big(\mathbf{r}_{il}\big) - \sum_{l\neq i,j} \nabla K_h\big(\mathbf{r}_{jl}\big) \right)$$

$$\begin{aligned} \text{s.t.} \qquad & \|\mathbf{r}_{ij}\|_2 \leq \|\mathbf{r}_{il}\|_2, && \forall (i,l) \\ & \|\mathbf{r}_{ij}\|_2 \leq \|\mathbf{r}_{jl}\|_2, && \forall (j,l) \end{aligned}$$

Proposition 4 *If the solution obtained from Problem 4 satisfy (4.14), then the solution obtained from Problem 3 also satisfies (4.14).*

Proof: Note that Problem 3 and Problem 4 has the same objective function. Given $\|\mathbf{r}_{ij}\|_2 = R_c$, let the optimal values for the objective function of these problems be as J_1^* and J_2^* respectively. Then it is known that $J_2^* \geq J_1^*$, since Problem 4 has less number of constraints, hence, larger feasible domain to optimize over. Let $\bar{\eta}_{ij}^{(1)}$ and $\bar{\eta}_{ij}^{(2)}$ are the evaluation of (4.16) for J_1^* and J_2^*, then is can be observed that,

$$\bar{\eta}_{ij}^{(1)} + \frac{J_1^*}{N} = \bar{\eta}_{ij}^{(2)} + \frac{J_2^*}{N} \geq \bar{\eta}_{ij}^{(2)} + \frac{J_1^*}{N},$$

and if $\bar{\eta}_{ij}^{(2)} \geq 0$, then $\bar{\eta}_{ij}^{(1)} \geq \bar{\eta}_{ij}^{(2)} \geq 0$. ■

4.4.1 KKT Conditions for Relaxed Problem

To identify the relaxed worst case configuration of agents in a swarm, we derive the Karush-Kuhn-Tucker (KKT) conditions of Problem 4. Define the Lagrangian as follows:

$$\begin{aligned} L(\mathbf{x}, \mathbf{u}, \mathbf{v}) \;&=\; \mathbf{r}_{ij}^{\mathsf{T}} \left(\sum_{l \neq i,j} \nabla K_h\big(\mathbf{r}_{il}\big) - \sum_{l \neq i,j} \nabla K_h\big(\mathbf{r}_{jl}\big) \right) \\ &+\; \sum_{l \neq i,j} \left(u_l \Big(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{il}\right\|^2 \Big) + v_l \Big(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{jl}\right\|^2 \Big) \right), \end{aligned}$$

the KKT conditions are:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{x}_l} L \;&=\; \Big(\nabla_{\mathbf{x}_l} \nabla K_h\big(\mathbf{r}_{il}\big) - \nabla_{\mathbf{x}_l} \nabla K_h\big(\mathbf{r}_{jl}\big) \Big)\, \mathbf{r}_{ij} \\ &\quad + 2u_l\big(\mathbf{r}_{il}\big) + 2v_l\big(\mathbf{r}_{jl}\big) \;=\; 0 \qquad \forall l \neq \{i,j\}, \end{aligned}$$

$$\left. \begin{aligned} u_l \Big(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{il}\right\|^2 \Big) \;&=\; 0 \\ v_l \Big(\left\|\mathbf{r}_{ij}\right\|^2 - \left\|\mathbf{r}_{jl}\right\|^2 \Big) \;&=\; 0 \\ u_l, v_l \;&\geq\; 0 \end{aligned} \right\} \quad \forall l \neq \{i,j\},$$

It is crucial to note that the equations for each agent $\mathbf{x}_l$ where $l \neq \{i,j\}$ are independent of each other. Furthermore, the cost function can be written as sum of sub-costs where each sub-cost represents the cost incurred by individual agent. Consequently, one can observe that using the *relaxed formulation* decomposes one optimization problem with $(N-2)d$ variables to $N-2$ number of optimization problems with d variables. Therefore, solving only 3-agent problem will suffice for further analysis, and this result can be used to prove collision avoidance for arbitrary N case. This relaxation will yield conservative results for systems where $N \geq 5$, but ensuring the collision avoidance using Problem 4, will guarantee collision avoidance property for (4.11) in arbitrary dimensions.

Now, using Definition 11 along with decoupling thanks to relaxation in Problem 4 one has

$$\begin{aligned}\bar{\eta}_{ij} &= -\frac{2}{N}\mathbf{r}_{ij}^{\mathsf{T}}\nabla K_h\big(\mathbf{r}_{ij}\big)\\ &\quad -\frac{N-2}{N}\max_{\mathbf{x}_l}\ \big\{\big\|\mathbf{r}_{ij}\big\|\big\|\nabla K_h\big(\mathbf{r}_{il}\big)-\nabla K_h\big(\mathbf{r}_{jl}\big)\big\|\big\}\\ &\quad -\big\|\mathbf{r}_{ij}\big\|\big\|\nabla\rho_D\big(\mathbf{x}_i\big)-\nabla\rho_D\big(\mathbf{x}_j\big)\big\|,\end{aligned}$$

using Assumption 2 and Lipschitz continuous gradient $\nabla\rho_D(\mathbf{x})$ yields,

$$\bar{\eta}_{ij} = -\frac{2}{N}\mathbf{r}_{ij}^{\mathsf{T}}\nabla K_h\big(\mathbf{r}_{ij}\big)-\frac{N-2}{N}{R_c}^2L_h-{R_c}^2L_{\rho_D}, \tag{4.23}$$

Now (4.23) will be investigated for both 1-dimensional kernel (4.1) and Laplacian kernel (4.2) for arbitrary dimensions.

4.4.2 Comparison of Exact and Conservative Analysis for 1-Dimensional Kernel

Substituting the gradient of (4.1) and L_h to (4.23) yields the following,

$$\begin{aligned}\bar{\eta}_{ij} &= -\frac{2}{N}\mathbf{r}_{ij}^{\mathsf{T}}\nabla K_h\big(\mathbf{r}_{ij}\big)-{R_c}^2\left(\frac{N-2}{N}L_h+L_{\rho_D}\right)\\ &= \frac{2}{N}\frac{R_ch}{(R_c+h)^3}-{R_c}^2\left(\frac{N-2}{N}\frac{3}{h^3}+L_{\rho_D}\right)\ \geq\ 0,\end{aligned}$$

with $N=2M$, further simplification yields,

$$\begin{aligned}\frac{h}{(R_c+h)^3}-R_c\left((M-1)\frac{3}{h^3}+ML_{\rho_D}\right)\ &\geq\ 0, \\ \frac{1}{(\xi+1)^3}-\xi\Big(3(M-1)+Mh^3L_{\rho_D}\Big)\ &\geq\ 0,\end{aligned}\tag{4.24}$$

where $\xi=R_c/h$. Note that $\big(3(M-1)+Mh^3L_{\rho_D}\big)>0$ for all $M>1$. Define the polynomial,

$$p(\xi)=\xi\big(\xi+1\big)^3-\frac{1}{3(M-1)+Mh^3L_{\rho_D}}\leq 0.$$

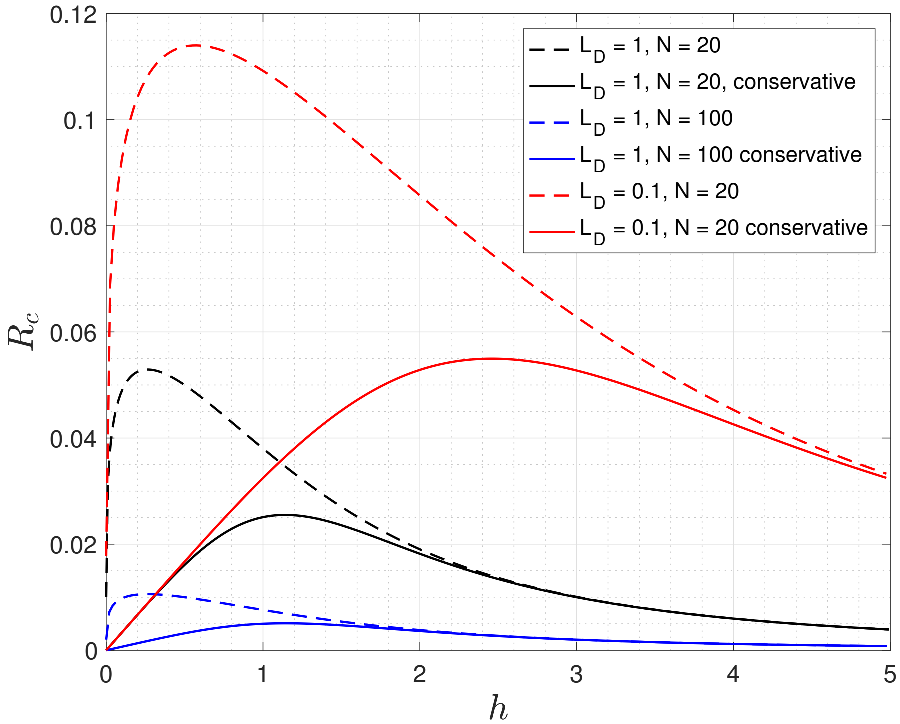

Figure 4.4: Exact (4.22) versus conservative (4.24) bounds on collision radius R_c with respect to interpolation length h for different number of agents N and desired density gradient Lipschitz constants L_{ρ_D}

which has all positive coefficients but the last one, hence, the number of variations in sign in the sequence of the coefficients is only one. Therefore, by *Descartes's Rule of Signs* [128], there is exactly one positive real root of $p(\xi) = 0$, i.e., $\exists \xi\big(M, h, L_{\rho_D}\big) > 0$, consequently,

$$R_c \leq \xi\big(M, h, L_{\rho_D}\big)h.$$

From exact (4.22) and conservative (4.24) bound inequalities, one can verify that the set of parameters R_c, h, M, L_{ρ_D} that satisfies (4.22) also satisfies (4.24). Figure 4.4 demonstrates the comparison of curves defined by (4.22) and (4.24).

4.4.3 Conservative Analysis for Laplacian Kernel

Similar to the previous analysis for 1-dimensional kernel case, substituting the gradient of (4.2) and L_h in (4.23) yields the following,

$$\begin{aligned}
\bar{\eta}_{ij} &= -\frac{2}{N}\mathbf{r}_{ij}^{\mathsf{T}}\nabla K_h\big(\mathbf{r}_{ij}\big) - {R_c}^2\left(\frac{N-2}{N}L_h + L_{\rho_D}\right) \\
&= \frac{2R_c}{NC_d h^{d+1}}\exp\left(-\frac{R_c}{h}\right) \\
&- {R_c}^2\left(\frac{N-2}{N}\frac{1}{C_d h^{d+2}} + L_{\rho_D}\right) \geq 0,
\end{aligned}$$

further simplification yields,

$$\begin{aligned}
\exp\left(-\frac{R_c}{h}\right) - R_c\left(\frac{M-1}{h} + C_d h^{d+1} M L_{\rho_D}\right) &\geq 0, \qquad (4.25)\\
\exp\big(-\xi\big) - \xi\Big((M-1) + C_d h^{d+2} M L_{\rho_D}\Big) &\geq 0,
\end{aligned}$$

where $\xi = R_c/h$ and $N = 2M$. Note that $\big((M-1) + C_d h^{d+2} M L_{\rho_D}\big) > 0$ for all $M > 1$.

For existence of collision radius $R_c > 0$, using the lower bound for $\exp(-\xi)$ gives,

$$\exp(-\xi) \geq 1-\xi \geq \xi\Big((M-1) + C_d h^{d+2} M L_{\rho_D}\Big)$$

and further manipulation of the inequality yields,

$$\begin{aligned}
1-\xi &\geq \xi\Big((M-1) + C_d h^{d+2} M L_{\rho_D}\Big) \\
1 &\geq \xi\Big(M + C_d h^{d+2} M L_{\rho_D}\Big)
\end{aligned}$$

which gives the positive upper bound $R_c > 0$ as follows,

$$\frac{h}{M(1 + C_d h^{d+2} L_{\rho_D})} \geq R_c. \qquad (4.26)$$

Note that conservatism in (4.26) is twofold: the conservatism introduced by relaxed problem, i.e., Problem 4, and the conservatism due to a lower bounding function for $\exp(-\xi)$. However, the upper bound (4.26), not only ensures collision avoidance but also gives a coarse method to choose interpolation length h. If one wants to eliminate the latter factor, the inequality (4.25) can be solved numerically. As it can be seen from Figure 4.5, the difference between

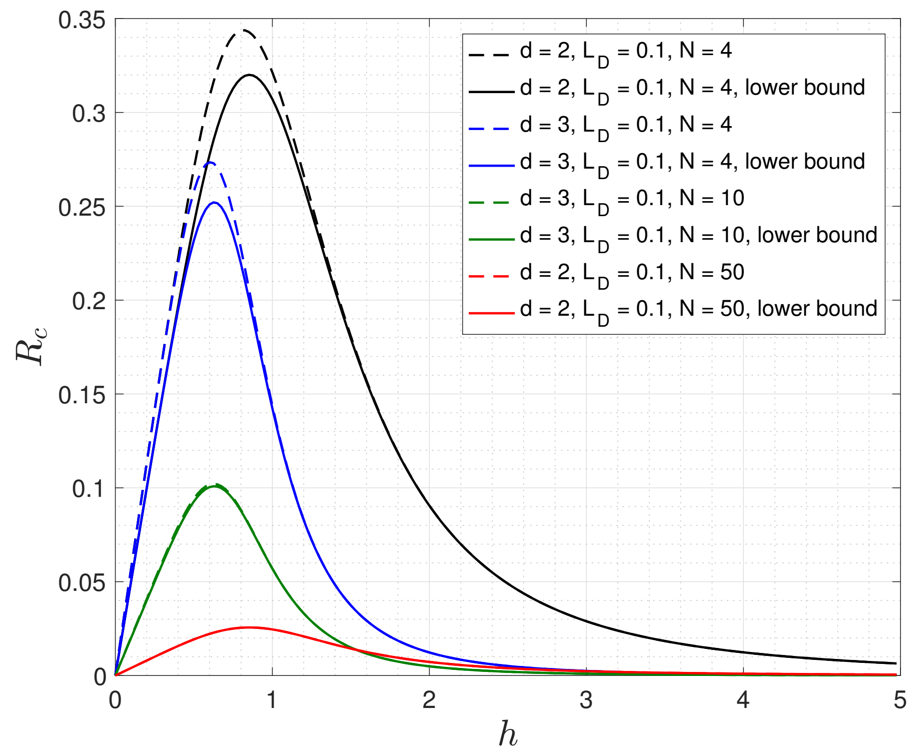

Figure 4.5: Comparison between upper bounds of collision avoidance radius R_c from (4.25) and (4.26) with respect to interpolation length h for different number of dimensions d, number of agents N and desired density gradient Lipschitz constants L_{ρ_D}

bounds obtained by (4.25) and (4.26) decreases as the number of agents increases.

Chapter 5

ADAPTIVE DENSITY ESTIMATION

In the preceding chapters, density-based swarm guidance and control were formulated using continuum representations, with kernel density estimation and heat-equation-driven dynamics serving as the primary instruments for shaping collective behavior. In these formulations, the smoothing parameter governing density estimation was treated as a design constant, selected through heuristics (3.21), global tuning considerations (3.5.1) or collision avoidance radius bounds (4.26). While effective in many scenarios, such fixed choices impose an implicit scale on the swarm, limiting responsiveness to spatial heterogeneity, local congestion, and evolving mission requirements. This chapter removes the smoothing parameter from the realm of heuristic design and instead embeds it directly into the dynamical structure of the system. By allowing the kernel bandwidth to evolve locally and autonomously, the swarm acquires the ability to adapt its own perceptual resolution in space and time. Importantly, this adaptation is achieved without introducing centralized coordination or explicit global optimization. Guided by the same diffusion principles used to regulate swarm density, each agent responds only to locally available density information and simple update laws, yet the resulting density estimate exhibits coherent, globally meaningful structure. This perspective aligns with the broader philosophy of behavior-based and emergent systems, wherein complex collective organization arises from the interaction of simple mechanisms rather than from elaborate agent-level intelligence. Here, adaptive density resolution emerges as a collective property of the swarm, not as an externally imposed design choice. The resulting framework preserves the analytical tractability of heat-equation-based density evolution while extending its expressiveness and robustness in dynamically changing environments. Simulation results demonstrate that locally adaptive density estimation improves responsiveness to spatial non-uniformities and enhances the performance of density-based control laws, particularly in regimes where fixed smoothing scales prove inadequate.

The chapter proceeds as follows. Notation is first established and properties of the heat semigroup and its kernel representation are recalled, providing the continuum foundation for density evolution. A coupled continuum formulation of interacting heat equations is then introduced, together with its associated semigroup and stability properties. Building on this, a finite-agent kernel density representation is derived, revealing an explicit connection between agent interactions and the underlying continuum diffusion structure. The adaptive bandwidth is not introduced as an independent modeling assumption, but instead arises from consistency between kernel smoothing and the underlying diffusion dynamics. This leads to a weak formulation of the resulting transport-diffusion system, clarifying how bandwidth evolution can be interpreted in a distributional sense. Finally, a decentralized implementation is developed in which both agent dynamics and adaptive bandwidth evolution are expressed in distributed form. Dynamic average consensus estimators are introduced to approximate the global functionals appearing in the bandwidth dynamics using only local neighbor-to-neighbor communication, yielding a fully distributed realization of the coupled system.

5.1 Background

The following notation and definitions are adopted for use throughout this chapter. The sets of natural numbers and positive natural numbers are denoted by $\mathbb{N}_0$, $\mathbb{N}$ respectively. The set of positive real numbers is denoted by $\mathbb{R}_+$. The closure of set Ω is denoted by $\overline{\Omega}$. The boundary $\partial\Omega$ of set Ω is defined as $\overline{\Omega} \setminus \Omega$ with $\setminus$ denoting set subtraction. The partial derivative with respect to the i^{th} variable is denoted by ∂_i. Partial derivatives of arbitrary order are denoted by $D^\alpha = \prod_{i=1}^d \partial_i^{\alpha_i}$, where $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$ is a multi-index. The notation $\bigoplus_{i=1}^N$ indicates the direct sum that consisting of N-component vectors with each component treated independently.

The analysis in this chapter relies on standard results from semigroup theory, Sobolev spaces, and Lyapunov stability summarized in Appendix D.

5.1.1 Heat Equation as a Generator of C_0-semigroup

Consider the heat equation:

$$\begin{aligned} \dot{\vartheta}(x,t) &= \alpha\Delta\vartheta(x,t), && \text{in} \quad \mathbb{R}^d \times (0,\infty) \\ \vartheta(x,0) &= \vartheta_0(x) && \text{on} \quad \mathbb{R}^d \end{aligned} \tag{5.1}$$

for $\alpha > 0$, where $\vartheta_0(x)$ is a given function in $X := L^p(\mathbb{R}^d),\ 1 \le p < \infty$. The solution to this heat equation is explicitly given by,

$$\big(S(t)\vartheta_0\big)(x) := \frac{1}{(4\pi\alpha t)^{d/2}} \int_{\mathbb{R}^d} e^{-\frac{\|x-\tau\|^2}{4\alpha t}} \vartheta_0(\tau)\, d\tau, \qquad t > 0,\ x \in \mathbb{R}^d, \tag{5.2}$$

where $\vartheta_0 \in L^p(\mathbb{R}^d)$. The definition (5.2) is known as *heat semigroup*, *Gaussian semigroup* or *diffusion semigroup*, and was one of the main sources for the development of semigroup theory [52]. By defining *heat kernel* as follows,

$$K_t(x) := \frac{1}{(4\pi\alpha t)^{d/2}}\, e^{-\frac{\|x\|^2}{4\alpha t}}, \qquad \int_{\mathbb{R}^d} K_t(x)\, dx = 1, \quad t > 0, \tag{5.3}$$

the heat semigroup can be restated in a convolution form as follows,

$$\big(S(t)\vartheta_0\big)(x) = \big(K_t * \vartheta_0\big)(x). \tag{5.4}$$

Note that $K_t \in \mathbb{S}(\mathbb{R}^d)$. Next step is to demonstrate that heat semigroup is a C_0-semigroup, and it is given by the following proposition detailed proof of which can be found in [74].

Proposition 5 *The family of maps $S(t)$ given by,*

$$\begin{aligned} &\big(S(t)f\big)(x) = \big(K_t * f\big)(x), && \text{for } t > 0, \\ &\big(S(0)f\big)(x) = f(x) \end{aligned}$$

is a strongly continuous semigroup (C_0-semigroup) on $\mathbb{S}(\mathbb{R}^d)$, furthermore its infinitesimal *generator is,*

$$A_S f = \lim_{t \to 0^+} \frac{1}{t}\big(S(t)f - f\big) = \alpha \Delta f,$$

for all $f \in \mathbb{S}(\mathbb{R}^d)$.

$\mathbb{S}(\mathbb{R}^d)$ is not a Banach space. Fortunately, the properties of heat semigroup has been studied in Banach spaces, i.e. Sobolev space, and the following results state the C_0-semigroup and its generator in $W^{k,p}(\mathbb{R}^d)$ and its detailed proof can be found in [89, 74].

Proposition 6 *The semigroup $\{S(t),\ t \geq 0\}$ is a C_0-semigroup on $W^{k,p}(\mathbb{R}^d)$ for $1 \leq p < \infty$ and $k \in \mathbb{N}_0$.*

Theorem 8 *The generator A_S of the semigroup $\{S(t),\ t \geq 0\}$ on $L^p(\mathbb{R}^d)$ for $1 < p < \infty$ is given by $A_S f = \alpha \Delta f$ and its domain $\mathcal{D}(A_S)$ is the Sobolev space $W^{2,p}(\mathbb{R}^d)$.*

Remark 10 *$\mathbb{S}(\mathbb{R}^d)$ is invariant under the semigroup $\{S(t),\ t \geq 0\}$ and dense both in $L^p(\mathbb{R}^d)$ and $W^{k,p}(\mathbb{R}^d)$ for all $1 \leq p < \infty$. Therefore, using the Definition 19 and Theorem 8, $\mathbb{S}(\mathbb{R}^d)$ is the core for generator $A_S : W^{2,p}(\mathbb{R}^d) \to L^p(\mathbb{R}^d)$ for $1 < p < \infty$.*

Corollary 6 *The heat equation given by (5.1) with $\Theta(0) = \vartheta_0 \in W^{2,p}(\mathbb{R}^d) \subset L^p(\mathbb{R}^d)$ has a unique solution $\Theta \in C^1\big(\mathbb{R}_+, L^p(\mathbb{R}^d)\big)$, where $1 < p < \infty$, and as it is previously stated by (5.2), the solution is $\Theta(t) = S(t)\vartheta_0$, $t > 0$.*

Theorem 9 (Green's Formula [54]) *Given U is a bounded, open subset of $\mathbb{R}^d$, and ∂U is C^1, let $u, v \in C^2(\bar{U})$. Then,*

$$\int_U \nabla v \cdot \nabla u \ dx \ = \ -\int_U u \Delta v \ dx \ + \ \int_{\partial U} (\nu \cdot \nabla v) u \ dS,$$

where $\nu \in \mathbb{R}^d$ is the outward pointing unit normal vector field on ∂U.

5.2 Problem Formulation in the Continuum

Consider the set of interacting heat equations as follows:

$$\left.\begin{aligned} \dot{\vartheta}_i - \alpha\Delta\vartheta_i &= -\beta\sum_{i\neq j}(\vartheta_i - \vartheta_j), \qquad \text{in} \quad \mathbb{R}^d \times (0,\infty) \\ \vartheta_i(x,0) &= \rho_i\big(x,0\big) - \rho_D(x) \qquad \text{on} \quad \mathbb{R}^d \end{aligned}\right\} \forall i \in \{1,\ldots,N\} \tag{5.5}$$

for $\alpha, \beta > 0$. Assume $\vartheta_i(x,0) \in W^{2,p}(\mathbb{R}^d)$, for all $i \in \{1,\ldots,N\}$. One can write the system of heat equations (5.5) in the following operator form as an abstract Cauchy problem:

$$\begin{aligned} \dot{\mathcal{V}}(x,t) &= \tilde{A}\mathcal{V}(x,t), \\ \mathcal{V}(x,0) &= \mathcal{V}_0(x), \end{aligned} \tag{5.6}$$

where $\mathcal{V}(x,t) := [\vartheta_1(x,t),\ldots,\vartheta_N(x,t)]^\mathsf{T} \in \bigoplus_{i=1}^N W^{2,p}(\mathbb{R}^d)$, and $\tilde{A} : \bigoplus_{i=1}^N W^{2,p}(\mathbb{R}^d) \to \bigoplus_{i=1}^N L^p(\mathbb{R}^d)$ is a linear operator defined as follows:

$$\tilde{A} = A - \beta L, \quad \text{where} \quad A = \alpha\bigoplus_{i=1}^N \Delta, \tag{5.7}$$

and $L \in \mathbb{R}^{N\times N} \succeq 0$ is the *Laplacian matrix* of the connected communication graph between individual systems. Throughout the chapter, the variables are related by $\vartheta_i(x,t) = \rho_i(x,t) - \rho_D(x)$. The next objective is to investigate the existence, uniqueness, and stability properties of the system (5.6).

Theorem 10 *Given a linear operator* $\tilde{A} : \bigoplus_{i=1}^N W^{2,p}(\mathbb{R}^d) \to \bigoplus_{i=1}^N L^p(\mathbb{R}^d)$ *such that*

$$\tilde{A} = \alpha\bigoplus_{i=1}^N \Delta - \beta L, \qquad \alpha, \beta > 0,$$

then the operator $\tilde{A}$ *is the infinitesimal generator of a* C_0*-semigroup* $\tilde{S}(t)$ *on* $\bigoplus_{i=1}^N L^p(\mathbb{R}^d)$ *for* $1 < p < \infty$.

Proof: From Theorem 8, we know that $\alpha\Delta\vartheta_i$ is the infinitesimal generator of a C_0-semigroup on $L^p(\mathbb{R}^d)$ for $1 < p < \infty$ and its domain is the Sobolev space $W^{2,p}(\mathbb{R}^d)$. Given

$$\tilde{A}\mathcal{V} = A\mathcal{V} - \beta L\mathcal{V} = \alpha \bigoplus_{i=1}^{N} \Delta\vartheta_i - \beta L\mathcal{V},$$

where the first part of the sum is the direct sum of infinitesimal generators of C_0-semigroups (heat semigroups), so it is itself the infinitesimal generator of a C_0-semigroup [77]. The second part of the sum is composed of a linear bounded operator, $\|\beta L\mathcal{V}\| \leq \beta(N-1)\|\mathcal{V}\|$, hence using the bounded perturbation Theorem 15 in the domain of $\bigoplus_{i=1}^{N} W^{2,p}(\mathbb{R}^d)$ with $1 < p < \infty$, it follows that the operator $\tilde{A}$ is the infinitesimal generator of a C_0-semigroup $\tilde{S}(t)$ on $\bigoplus_{i=1}^{N} L^p(\mathbb{R}^d)$. ■

Corollary 7 (Existence & Uniqueness) *Let $1 < p < \infty$, then the heat equations system given by (5.5), with initial condition $\mathcal{V}_0 \in \bigoplus_{i=1}^{N} W^{2,p}(\mathbb{R}^d)$ has a unique solution $\mathcal{V} \in C^1(\mathbb{R}_+, \bigoplus_{i=1}^{N} L^p(\mathbb{R}^d))$ and it is explicitly given by $\mathcal{V}(t) = \tilde{S}(t)\mathcal{V}_0,\ t \geq 0$, for any $\mathcal{V}_0 \in \bigoplus_{i=1}^{N} W^{2,p}(\mathbb{R}^d)$.*

Proof: Corollary is the direct result of Theorem 16, and Theorem 10. ■

5.2.1 Continuum Stability Analysis

Having established existence and uniqueness of solutions for the coupled heat system (5.5), attention is now directed toward its stability and asymptotic convergence properties. Consider *density error cost* given as follows:

$$\Phi(t) \;=\; \frac{1}{2}\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}}\mathcal{V}\ dx \;=\; \frac{1}{2}\sum_{i=1}^{N}\int_{\mathbb{R}^d} \vartheta_i^2\ dx \;=\; \frac{1}{2}\sum_{i=1}^{N}\int_{\mathbb{R}^d} \big(\rho_i(x,t) - \rho_D(x)\big)^2\ dx \qquad (5.8)$$

The following result establishes that the density error functional (5.8) is non-increasing under the heat system dynamics (5.5).

Lemma 9 (Non-increasing density error) *Let* $\rho_D, \rho_i\big(x,t\big) \in \mathbb{S}(\mathbb{R}^d)$ $\forall i \in \{1,\ldots,N\}$, *and* $L \succeq 0$, *then density error given by (5.8) under the dynamics (5.5) is non-increasing, i.e.,* $\dot{\Phi}(t) \leq 0$.

Proof:

$$\begin{aligned}
\frac{d}{dt}\Phi \;&=\; \frac{d}{dt}\left(\frac{1}{2}\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}}\mathcal{V}\;dx\right) \;=\; \int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}}\dot{\mathcal{V}}\;dx \;=\; \int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}}\tilde{A}\mathcal{V}\;dx \;=\; \int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}}\big(A-\beta L\big)\mathcal{V}\;dx \\
&=\; \int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} A\mathcal{V}\;dx \;-\; \beta\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L\mathcal{V}\;dx, \\
&=\; \alpha\sum_{i=1}^{N}\int_{\mathbb{R}^d} \vartheta_i\,\Delta\vartheta_i\;dx \;-\; \beta\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L\mathcal{V}\;dx
\end{aligned}$$

and the first term can be written as,

$$\begin{aligned}
\int_{\mathbb{R}^d} \vartheta_i\,\Delta\vartheta_i\;dx \;&=\; \lim_{R\to\infty}\int_{\mathcal{B}_R(0)} \vartheta_i\,\Delta\vartheta_i\;dx \\
&=\; -\lim_{R\to\infty}\int_{\mathcal{B}_R(0)} \|\nabla\vartheta_i\|^2\;dx \;+\; \lim_{R\to\infty}\int_{\partial\mathcal{B}_R(0)} \vartheta_i\,(\nu\cdot\nabla\vartheta_i)\;dS \quad \text{(via Theorem 9)} \\
&=\; -\int_{\mathbb{R}^d} \|\nabla\vartheta_i\|^2\;dx.
\end{aligned}$$

The last equality is due to the fact that $\rho_D, \rho_i\big(x,t\big) \in \mathbb{S}(\mathbb{R}^d)$, hence, $\vartheta_i \in \mathbb{S}(\mathbb{R}^d)$ and $\lim_{\|x\|\to\infty}\|x\|^k D^\alpha\vartheta_i = 0$, for all $k \in \mathbb{N}_0$ and $\alpha \in \mathbb{N}_0^d$, i.e., ϑ_i and its partial derivatives decay rapidly ($o(R^{-k})$ $\forall k \in \mathbb{N}_0$), while the surface measure only scales with $o(R^{d-1})$. Therefore the surface integral on $\partial\mathcal{B}_R(0)$ goes to zero as $R \to \infty$. Substituting it back to time derivative yields,

$$\begin{aligned}
\frac{d}{dt}\Phi \;&=\; -\alpha\sum_{i=1}^{N}\int_{\mathbb{R}^d} \|\nabla\vartheta_i\|^2\,dx - \beta\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L\mathcal{V}\;dx \\
&=\; -\left(\alpha\sum_{i=1}^{N}\int_{\mathbb{R}^d} \|\nabla\vartheta_i\|^2\;dx \;+\; \beta\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L\mathcal{V}\;dx\right) \;\leq\; 0.
\end{aligned}$$

■

The next result demonstrates the convexity of density error (5.8) with respect to time argument under the heat system dynamics (5.5).

Lemma 10 (Convexity of density error in time) *Density error $\Phi(t)$ given by (5.8) is convex with respect to $t \in \mathbb{R}_+$ under the heat system dynamics (5.5).*

Proof: It suffices to check the second derivative of the density error in time:

$$
\begin{aligned}
\frac{d^2}{dt^2}\Phi &= -\frac{d}{dt}\left(\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} \|\nabla \vartheta_i\|^2 \, dx \; + \; \beta \int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L \mathcal{V} \, dx\right) \\
&= -2\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} (\nabla \vartheta_i)^{\mathsf{T}} \nabla \dot{\vartheta}_i \, dx \; - \; 2\beta \int_{\mathbb{R}^d} \dot{\mathcal{V}}^{\mathsf{T}} L \mathcal{V} \, dx \qquad \text{using Leibniz rule over } \mathbb{S}(\mathbb{R}^d) \\
&= 2\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} \dot{\vartheta}_i \, \Delta \vartheta_i \, dx \; - \; 2\beta \int_{\mathbb{R}^d} \dot{\mathcal{V}}^{\mathsf{T}} L \mathcal{V} \, dx \qquad \text{using Theorem (9)} \\
&= 2 \int_{\mathbb{R}^d} \dot{\mathcal{V}}^{\mathsf{T}} \left(\alpha \bigoplus_{i=1}^{N} \Delta \vartheta_i\right) dx \; - \; 2 \int_{\mathbb{R}^d} \dot{\mathcal{V}}^{\mathsf{T}} (\beta L \mathcal{V}) \, dx \\
&= 2 \int_{\mathbb{R}^d} \dot{\mathcal{V}}^{\mathsf{T}} \dot{\mathcal{V}} \, dx \; = \; 2\|\dot{\mathcal{V}}\|^2_{L_2(\mathbb{R}^d)} \; \geq \; 0.
\end{aligned}
$$

■

Thus the density error evolves as a convex, non-increasing functional in time, with its rate of decrease approaching zero monotonically. Using Lemma 9, 10, one can argue that $\lim_{t\to\infty} \Phi(t) = c$ and $c \geq 0$. The following is the formal argument for *Asymptotic Stability* of the heat system dynamics (5.5).

Theorem 11 (Asymptotic Stability) *Given $\rho_D, \rho_i \in \mathbb{S}(\mathbb{R}^d)$ $\forall i \in \{1, \ldots, N\}$, consider the system of heat equations (5.5) in abstract Cauchy form $\dot{\mathcal{V}} = \tilde{A}\mathcal{V}$ with initial condition $\mathcal{V}_0 \in \bigoplus_{i=1}^{N} \mathbb{S}(\mathbb{R}^d) \subset \bigoplus_{i=1}^{N} W^{2,p}(\mathbb{R}^d)$. Let $\tilde{A} : \bigoplus_{i=1}^{N} W^{2,p}(\mathbb{R}^d) \to \bigoplus_{i=1}^{N} L^p(\mathbb{R}^d)$ for $1 < p < \infty$ denote the infinitesimal generator of the C_0-semigroup $\{\tilde{S}(t),\; t \geq 0\}$. Then each orbit $\Upsilon\big(\mathcal{V}_0\big) = \{\tilde{S}(t)\mathcal{V}_0,\; t \geq 0\}$ of the system (5.5) is bounded in $\bigoplus_{i=1}^{N} L_2(\mathbb{R}^d)$, and each agent's density error vanishes asymptotically in $L_2(\mathbb{R}^d)$, i.e., $\lim_{t\to\infty} \|\rho_i(\cdot, t) - \rho_D\|_{L_2(\mathbb{R}^d)} = 0$ $\forall i \in \{1, \ldots, N\}$.*

Proof: To establish stability, we consider the following candidate Lyapunov functional,

$$V\big(\mathcal{V}\big) = \frac{\alpha}{2}\sum_{i=1}^{N}\int_{\mathbb{R}^d} \|\nabla\vartheta_i\|^2\,dx + \frac{\beta}{2}\int_{\mathbb{R}^d} \mathcal{V}^{\mathsf{T}} L\mathcal{V}\,dx,$$

which is non-negative, and taking the time derivative along the trajectories of the system yields $\dot{V}\big(\mathcal{V}\big) = -\|\dot{\mathcal{V}}\|^2_{L_2(\mathbb{R}^d)} \leq 0$ by Lemma 10, confirming that $V(\mathcal{V}(t)) \leq V(\mathcal{V}_0)\ \ \forall t \geq 0$. Moreover, by Lemma 9, $\Phi(t) = \frac{1}{2}\|\mathcal{V}(\cdot,t)\|^2_{L_2(\mathbb{R}^d)} \leq \Phi(0)$, $\forall t \geq 0$, hence $\|\mathcal{V}(\cdot,t)\|_{L_2(\mathbb{R}^d)} \leq \sqrt{2\Phi(0)}$. Therefore the positive orbit $\Upsilon(\mathcal{V}_0) = \{\tilde{S}(t)\mathcal{V}_0 : t \geq 0\}$ is bounded in $(\bigoplus_{i=1}^{N} L_2(\mathbb{R}^d))$.

To establish asymptotic convergence to the equilibrium point, we explicitly evaluate the action of the C_0-semigroup $\tilde{S}(t) = e^{\tilde{A}t}$ via the spatial Fourier transform. Let $\widehat{\mathcal{V}}(\xi,t)$ denote the component-wise spatial Fourier transform of $\mathcal{V}(x,t)$ with respect to the frequency vector $\xi \in \mathbb{R}^d$. Transforming the abstract Cauchy form $\dot{\mathcal{V}} = \tilde{A}\mathcal{V} = \big(\alpha\bigoplus_{i=1}^{N}\Delta - \beta L\big)\mathcal{V}$ yields a system of decoupled linear matrix ordinary differential equations for each frequency ξ:

$$\frac{d}{dt}\widehat{\mathcal{V}}(\xi,t) = -\big(\alpha\|\xi\|^2 I_N + \beta L\big)\widehat{\mathcal{V}}(\xi,t).$$

Since the identity matrix I_N and the graph Laplacian L commute, the unique solution in the frequency domain is given explicitly by:

$$\widehat{\mathcal{V}}(\xi,t) = e^{-\alpha\|\xi\|^2 t} e^{-\beta L t}\widehat{\mathcal{V}}_0(\xi).$$

Using the spectral decomposition of the symmetric, positive semi-definite undirected connected graph Laplacian L with eigenvalues $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$ and corresponding orthonormal eigenvectors $\{\mathbf{v}_1,\ldots,\mathbf{v}_N\}$, where $\mathbf{v}_1 = \frac{1}{\sqrt{N}}\mathbf{1}$, the matrix exponential factors as:

$$e^{-\beta L t} = \mathbf{v}_1\mathbf{v}_1^{\mathsf{T}} + \sum_{k=2}^{N} e^{-\beta\lambda_k t}\mathbf{v}_k\mathbf{v}_k^{\mathsf{T}}.$$

Substituting this representation back into the Fourier solution yields:

$$\widehat{\mathcal{V}}(\xi,t) = e^{-\alpha\|\xi\|^2 t}\left(\mathbf{v}_1\mathbf{v}_1^\mathsf{T}\widehat{\mathcal{V}}_0(\xi) + \sum_{k=2}^{N} e^{-\beta\lambda_k t}\mathbf{v}_k\mathbf{v}_k^\mathsf{T}\widehat{\mathcal{V}}_0(\xi)\right).$$

By *Parseval-Plancherel identity*, the $L_2(\mathbb{R}^d)$ norm of the state trajectory satisfies:

$$\|\mathcal{V}(\cdot,t)\|^2_{L_2(\mathbb{R}^d)} = \int_{\mathbb{R}^d} e^{-2\alpha\|\xi\|^2 t}\left\|\mathbf{v}_1\mathbf{v}_1^\mathsf{T}\widehat{\mathcal{V}}_0(\xi) + \sum_{k=2}^{N} e^{-\beta\lambda_k t}\mathbf{v}_k\mathbf{v}_k^\mathsf{T}\widehat{\mathcal{V}}_0(\xi)\right\|^2 d\xi.$$

For any fixed $\xi \in \mathbb{R}^d \setminus \{0\}$, the continuous dispersion factor satisfies $\lim_{t\to\infty} e^{-2\alpha\|\xi\|^2 t} = 0$. Concurrently, for the discrete network topologies, $\lim_{t\to\infty} e^{-\beta\lambda_k t} = 0$ exponentially fast for all $k \geq 2$ since $\lambda_k > 0$ for a connected graph. Because $\mathcal{V}_0 \in \bigoplus_{i=1}^{N} \mathbb{S}(\mathbb{R}^d)$, its Fourier transform decays rapidly, ensuring that the integrand is bounded by an integrable function independent of t. Applying the *Lebesgue Dominated Convergence Theorem*, and observing that the integrand converges pointwise to zero for almost every $\xi \in \mathbb{R}^d$, allows the limit to be passed under the integral sign:

$$\begin{aligned}\lim_{t\to\infty}\|\mathcal{V}(\cdot,t)\|^2_{L_2(\mathbb{R}^d)} &= \int_{\mathbb{R}^d}\left(\lim_{t\to\infty} e^{-2\alpha\|\xi\|^2 t}\left\|\mathbf{v}_1\mathbf{v}_1^\mathsf{T}\widehat{\mathcal{V}}_0(\xi) + \sum_{k=2}^{N} e^{-\beta\lambda_k t}\mathbf{v}_k\mathbf{v}_k^\mathsf{T}\widehat{\mathcal{V}}_0(\xi)\right\|^2\right) d\xi \\ &= 0.\end{aligned}$$

Thus, $\lim_{t\to\infty}\|\mathcal{V}(\cdot,t)\|_{L_2(\mathbb{R}^d)} = 0$, establishing that the density error vector $\mathcal{V}(t)$ converges to zero asymptotically in the $L_2(\mathbb{R}^d)$ norm space, which implies the collective swarm profiles ρ_i converge to the target distribution ρ_D in the mean-square sense. ■

The following differential inequality further characterizes the dissipation structure and decay behavior of the system, verifying that the asymptotic convergence is algebraic rather than uniformly exponential. Starting with the fundamental dissipation relation:

$$\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} \|\nabla \vartheta_i\|^2 \, dx + \beta \int_{\mathbb{R}^d} \mathcal{V}^\mathsf{T} L \mathcal{V} \, dx = - \int_{\mathbb{R}^d} \mathcal{V}^\mathsf{T} \big(A - \beta L\big) \mathcal{V} \, dx,$$

and applying the Cauchy-Schwarz inequality to the right-hand side yields:

$$\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} \|\nabla \vartheta_i\|^2 \, dx + \beta \int_{\mathbb{R}^d} \mathcal{V}^\mathsf{T} L \mathcal{V} \, dx \leq \left(\int_{\mathbb{R}^d} \|\mathcal{V}\|^2 \, dx \right)^{1/2} \left(\int_{\mathbb{R}^d} \|A\mathcal{V} - \beta L \mathcal{V}\|^2 \, dx \right)^{1/2}$$

$$\left(\alpha \sum_{i=1}^{N} \int_{\mathbb{R}^d} \|\nabla \vartheta_i\|^2 \, dx + \beta \int_{\mathbb{R}^d} \mathcal{V}^\mathsf{T} L \mathcal{V} \, dx \right)^2 \leq \left(\int_{\mathbb{R}^d} \mathcal{V}^\mathsf{T} \mathcal{V} \, dx \right) \left(\int_{\mathbb{R}^d} \dot{\mathcal{V}}^\mathsf{T} \dot{\mathcal{V}} \, dx \right).$$

Using (5.8), the inequality can be framed in terms of the Lyapunov functional $V(\mathcal{V})$ and its derivatives:

$$\big(2V(\mathcal{V})\big)^2 \leq \big(2\Phi(t)\big) \left(-\dot{V}(\mathcal{V})\right),$$

which reduces to the non-linear differential inequality:

$$\dot{V}(\mathcal{V}) \leq -\frac{2}{\Phi(t)} V^2(\mathcal{V}). \tag{5.9}$$

Using Lemma 9, the functional $\Phi(t)$ is monotonically non-increasing and is bounded from above by its initial value, $\Phi(t) \leq \Phi(0) = \Phi_0$. Substituting this upper bound into the denominator of (5.9) preserves the inequality:

$$\dot{V}(t) \leq -\frac{2}{\Phi_0} V^2(t).$$

Separating variables and integrating from 0 to t yields:

$$-\int_{V(0)}^{V(t)} \frac{dV}{V^2} \geq \frac{2}{\Phi_0} \int_0^t dt \implies \frac{1}{V(t)} - \frac{1}{V(0)} \geq \frac{2}{\Phi_0} t.$$

Rearranging for $V(t)$ yields the explicit contractive bound:

$$V(t) \leq \frac{V(0)}{1 + \left(\frac{2V(0)}{\Phi_0}\right) t} = \mathcal{O}\left(t^{-1}\right).$$

This explicitly establishes an algebraic (polynomial) upper decay bound of order $\mathcal{O}(t^{-1})$. Also note that, the case of $\Phi(t) = 0$ is not excluded by Definition 23.

Remark 11 *The* LaSalle Invariance Principle *stated in Theorem [18] cannot be invoked directly in Theorem 11 because the spatial domain $\mathbb{R}^d$ is unbounded. In particular, bounded subsets of spaces such as $W^{2,p}(\mathbb{R}^d)$ or $L_2(\mathbb{R}^d)$ are generally not relatively compact, and therefore bounded trajectories do not necessarily possess precompact positive orbits. In this case, the existence of a non-increasing* Lyapunov *functional alone is insufficient to guarantee convergence of trajectories to the largest invariant subset of $\mathcal{V} : \dot{V}(\mathcal{V}) = 0$. This contrasts with the finite-dimensional case, where bounded sets are automatically relatively compact by the Heine–Borel theorem, allowing bounded trajectories to admit convergent subsequences. As a result, the standard finite-dimensional* LaSalle Invariance Principle *yields asymptotic convergence under substantially weaker compactness assumptions than those available in infinite-dimensional evolution equations posed on unbounded spatial domains.*

5.3 Finite-Agent Realization

To this point, the analysis has been conducted at the continuum level, where swarm evolution is represented by density fields rather than discrete agent trajectories. This section develops the corresponding finite-agent realization for the case of constant and identical kernel bandwidth parameters h across the swarm, establishing the connection between the continuum heat-equation formulation and distributed agent interactions. The resulting framework will subsequently serve as the foundation for the adaptive and spatially varying bandwidth dynamics introduced in the following section. Recall the following density estimate:

$$\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \frac{1}{N}\sum_{j=1}^{N} K_h\big(\mathbf{x}-\mathbf{x}_j\big) \;=\; \frac{1}{Nh^d}\sum_{j=1}^{N} K\left(\frac{\mathbf{x}-\mathbf{x}_j}{h}\right) \tag{5.10}$$

and its gradient at any point $\mathbf{x} \in \mathbb{R}^d$:

$$\nabla_{\mathbf{x}}\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \frac{1}{N}\sum_{j=1}^{N} \nabla_{\mathbf{x}} K_h\big(\mathbf{x}-\mathbf{x}_j\big) \;=\; \frac{1}{Nh^d}\sum_{j=1}^{N} \nabla_{\mathbf{x}} K\left(\frac{\mathbf{x}-\mathbf{x}_j}{h}\right) \tag{5.11}$$

where $\mathcal{X} := [\mathbf{x}_1^\mathsf{T}, \ldots, \mathbf{x}_N^\mathsf{T}]^\mathsf{T}$, i.e. the direct sum of location of all agents in the swarm. Similarly, density estimates and their gradients at agent locations $\mathbf{x}_i$ are defined as follows:

$$\rho_i \;:=\; \rho(\mathbf{x}_i, \mathcal{X}, h) \;=\; \frac{1}{Nh^d}\sum_{j=1}^{N} K\left(\frac{\mathbf{x}_i-\mathbf{x}_j}{h}\right), \tag{5.12}$$

$$\nabla\rho_i \;:=\; \nabla\rho(\mathbf{x}_i, \mathcal{X}, h) \;=\; \frac{1}{Nh^d}\sum_{j\neq i}^{N} \nabla K\left(\frac{\mathbf{x}_i-\mathbf{x}_j}{h}\right), \tag{5.13}$$

Recall the heat-equation-based control law that has been used for each agent $\mathbf{x}_i$ so far:

$$\dot{\mathbf{x}}_i = -\alpha\,\frac{\nabla\rho_i - \nabla\hat{\rho}_D(\mathbf{x}_i)}{\rho_i}, \tag{5.14}$$

where $\hat{\rho}_D(\mathbf{x})$ is the smoothed desired density (with an arbitrary choice of smoothing kernel) given as follows:

$$\hat{\rho}_D(\mathbf{x}) \;=\; (K_h * \rho_D)(\mathbf{x}). \tag{5.15}$$

Note that, so far the same kernel function $K_h(\mathbf{x})$ has been used to smoothen the desired density function $\rho_D(\mathbf{x})$. Now consider the previously defined (5.8) *density error cost* for the case of identical and constant, i.e., $h_j = h,\ \forall j \in \{1, \ldots, N\}$ and $\dot{h} = 0$ over the swarm as follows:

$$\min_{\mathcal{X}} \ \Phi(\mathcal{X}, h) \ = \ \min_{\mathcal{X}} \ \frac{1}{2} \sum_{j=1}^{N} \int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h_j) - \rho_D(\mathbf{x})\big)^2 \ \mathrm{d}\mathbf{x} \ = \ \min_{\mathcal{X}} \ \frac{N}{2} \int_{\mathbb{R}^d} \vartheta^2(\mathbf{x}, \mathcal{X}, h) \ \mathrm{d}\mathbf{x}. \tag{5.16}$$

One can calculate the gradient of the cost function $\Phi(\mathcal{X}, h)$ with respect the agent states $\mathbf{x}_i$ as follows:

$$\begin{aligned} \nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h) \ &= \ N \int_{\mathbb{R}^d} \vartheta(\mathbf{x}, \mathcal{X}, h) \ \nabla_{\mathbf{x}_i} \vartheta(\mathbf{x}, \mathcal{X}, h) \ \mathrm{d}\mathbf{x}, \qquad\qquad \forall i \in \{1, \dots, N\}, \\ &= \ N \left(\int_{\mathbb{R}^d} \rho(\mathbf{x}, \mathcal{X}, h) \ \nabla_{\mathbf{x}_i} \rho(\mathbf{x}, \mathcal{X}, h) \ \mathrm{d}\mathbf{x} \ - \int_{\mathbb{R}^d} \rho_D(\mathbf{x}) \ \nabla_{\mathbf{x}_i} \rho(\mathbf{x}, \mathcal{X}, h) \ \mathrm{d}\mathbf{x} \right), \end{aligned}$$

now using the following partial,

$$\nabla_{\mathbf{x}_i} \rho(\mathbf{x}, \mathcal{X}, h) = \frac{1}{N} \nabla_{\mathbf{x}_i} K_h\big(\mathbf{x} - \mathbf{x}_i\big), \qquad \forall x_i \in \mathcal{S}.$$

we have,

$$\nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h) \ = \ \int_{\mathbb{R}^d} \rho(\mathbf{x}, \mathcal{X}, h) \ \nabla_{\mathbf{x}_i} K_h\big(\mathbf{x} - \mathbf{x}_i\big) \ \mathrm{d}\mathbf{x} \ - \ \int_{\mathbb{R}^d} \rho_D(\mathbf{x}) \ \nabla_{\mathbf{x}_i} K_h\big(\mathbf{x} - \mathbf{x}_i\big) \ \mathrm{d}\mathbf{x}$$

assuming radially symmetric kernel function $K_h(\mathbf{x})$ implies $\nabla_{\mathbf{x}_i} K_h\big(\mathbf{x} - \mathbf{x}_i\big) = -\nabla_{\mathbf{x}_i} K_h\big(\mathbf{x}_i - \mathbf{x}\big) = \nabla_{\mathbf{x}} K_h\big(\mathbf{x}_i - \mathbf{x}\big)$, and yields the following,

$$\begin{aligned} \nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h) \ &= \ \int_{\mathbb{R}^d} \rho(\mathbf{x}, \mathcal{X}, h) \ \nabla_{\mathbf{x}} K_h\big(\mathbf{x}_i - \mathbf{x}\big) \ \mathrm{d}\mathbf{x} \ - \int_{\mathbb{R}^d} \rho_D(\mathbf{x}) \ \nabla_{\mathbf{x}} K_h\big(\mathbf{x}_i - \mathbf{x}\big) \ \mathrm{d}\mathbf{x} \\ &= \ \big(\rho * \nabla K_h\big)(\mathbf{x}_i) \ - \ \big(\rho_D * \nabla K_h\big)(\mathbf{x}_i). \end{aligned}$$

As long as $\rho_D(\mathbf{x})$ is integrable and $K_h(\mathbf{x}) \in C^1(\mathbb{R}^d, \mathbb{R})$ we have the following,

$$\begin{aligned}
\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h) &= \big(\rho * \nabla K_h\big)(\mathbf{x}_i) \;-\; \big(\rho_D * \nabla K_h\big)(\mathbf{x}_i), \\
&= \frac{1}{N}\sum_{j\neq i}^{N} \nabla\big(K_h * K_h\big)\big(\mathbf{x}_i - \mathbf{x}_j\big) \;-\; \nabla\big(\rho_D * K_h\big)(\mathbf{x}_i)
\end{aligned} \tag{5.17}$$

Considering the cost gradient calculated as (5.17) and defining a new kernel function as follows:

$$\hat{K}(\mathbf{x}) = \big(K_h * K_h\big)(\mathbf{x}), \tag{5.18}$$

and associated density estimates using $\hat{K}(\mathbf{x})$,

$$\begin{aligned}
\rho(\mathbf{x}_i) &= \frac{1}{N}\sum_{j=1}^{N} \hat{K}\left(\mathbf{x}_i - \mathbf{x}_j\right), \\
\nabla\rho(\mathbf{x}_i) &= \frac{1}{N}\sum_{j\neq i}^{N} \nabla\hat{K}\left(\mathbf{x}_i - \mathbf{x}_j\right), \\
\hat{\rho}_D(\mathbf{x}_i) &= (K_h * \rho_D)(\mathbf{x}_i),
\end{aligned}$$

the heat equation based control law given by (5.14) can be rewritten as:

$$\dot{\mathbf{x}}_i = -\alpha\frac{\nabla\rho\big(\mathbf{x}_i\big) - \nabla\hat{\rho}_D(\mathbf{x}_i)}{\rho\big(\mathbf{x}_i\big)} = -\alpha\frac{\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h)}{\rho\big(\mathbf{x}_i\big)}. \tag{5.19}$$

The expression (5.19) demonstrates the fact that, the heat equation based control law can be written in terms of gradient of the density error minimization cost and current calculated density for the finite number of agents case. The stability and convergence arguments for the global constant h in the finite agent case can be found in Appendix B.2.

Remark 12 *In case of using Gaussian Kernel for density estimations, i.e.,*

$$K_h(\mathbf{x}) \;=\; \frac{1}{\left(h\sqrt{\pi}\right)^d} \exp\left(-\frac{\mathbf{x}^\top \mathbf{x}}{h^2}\right)$$

self-convolution $(K_h * K_h)(\mathbf{x})$ *is also a Gaussian kernel with the following relation*

$$\hat{K}(\mathbf{x}) \;=\; \big(K_h * K_h\big)(\mathbf{x}) \;=\; \frac{1}{\left(h\sqrt{2\pi}\right)^d} \exp\left(-\frac{\mathbf{x}^\top \mathbf{x}}{2h^2}\right) \;=\; K_{\sqrt{2}h}(\mathbf{x}).$$

The preceding continuum analysis and finite-agent formulation establish the analytical and computational framework used to interpret adaptive bandwidth evolution as a diffusion-driven process. In particular, the heat-equation formulation provides the continuum dynamics governing density evolution, while the finite-agent realization connects these dynamics to distributed kernel-based interactions implementable through local agent communication. Building upon this foundation, the following section introduces adaptive kernel bandwidth dynamics, where each agent evolves its own local estimation scale according to diffusion-inspired update laws derived from the structure of the Gaussian heat kernel.

5.4 Individual & Adaptive Estimation

This section presents a method to find optimal parameters for the individual density estimates. Each agent calculates its own density estimation parameters, then using consensus, the swarm synchronizes the estimation parameters h_i to reach a global estimation parameter. In this section, the Gaussian kernel is assumed for all density estimations due to its relevance to the *heat kernel*,

$$K_h(\mathbf{x}) \;=\; \frac{1}{\left(h\sqrt{\pi}\right)^d} \exp\left(-\frac{\mathbf{x}^\top \mathbf{x}}{h^2}\right).$$

hence the density at any point $\mathbf{x} \in \mathbb{R}^d$ for N agent swarm is written as:

$$\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \frac{1}{N}\sum_{j=1}^{N} K_h(\mathbf{x}-\mathbf{x}_j) \;=\; \frac{1}{N\pi^{d/2}} \;\sum_{j=1}^{N} \frac{1}{h^d} \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_j\|^2}{h^2}\right).$$

The control law for the adaptive h_i parameters will be driven via heat equation system (5.5) whose stability has been proven in the previous section. The key observation enabling adaptive bandwidth dynamics is that, for Gaussian kernels, differentiation with respect to bandwidth is equivalent to diffusion in space:

Observation 1 *Consider the following derivations for Gaussian Kernel,*

$$\begin{aligned}
\frac{\partial}{\partial h}\rho(\mathbf{x}, \mathcal{X}, h) \;&=\; \frac{1}{N\pi^{d/2}} \;\sum_{j=1}^{N} \left(\frac{2\|\mathbf{x}-\mathbf{x}_j\|^2 - h^2\, d}{h^{d+3}}\right) \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_j\|^2}{h^2}\right) \\
&=\; \frac{1}{N}\sum_{j=1}^{N} \left(\frac{2\|\mathbf{x}-\mathbf{x}_j\|^2 - h^2\, d}{h^{3}}\right) K_h(\mathbf{x}-\mathbf{x}_j) \\
\Delta_{\mathbf{x}}\rho(\mathbf{x}, \mathcal{X}, h) \;&=\; \frac{2}{N\pi^{d/2}} \;\sum_{j=1}^{N} \left(\frac{2\|\mathbf{x}-\mathbf{x}_j\|^2 - h^2\, d}{h^{d+4}}\right) \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_j\|^2}{h^2}\right) \\
&=\; \frac{2}{N}\sum_{j=1}^{N} \left(\frac{2\|\mathbf{x}-\mathbf{x}_j\|^2 - h^2\, d}{h^{4}}\right) K_h(\mathbf{x}-\mathbf{x}_j)
\end{aligned}$$

which yields the following relation:

$$\frac{2}{h}\frac{\partial}{\partial h}\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \Delta_{\mathbf{x}}\rho(\mathbf{x}, \mathcal{X}, h). \tag{5.20}$$

Observation 1 provides the key structural bridge between adaptive density estimation and diffusion dynamics. As a consistency check, assume that $\dot{\mathbf{x}}_i = 0 \;\; \forall i$,

$$\dot{\rho}(\mathbf{x}, \mathcal{X}, h) \;=\; \dot{h}\,\frac{\partial}{\partial h}\rho(\mathbf{x}, \mathcal{X}, h)$$

and the heat equation becomes,

$$\dot{\rho}(\mathbf{x}, \mathcal{X}, h) \;=\; \dot{h}\,\frac{\partial}{\partial h}\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \Delta\rho(\mathbf{x}, \mathcal{X}, h) - \Delta\rho_D(\mathbf{x}),$$

and using (5.20) yields

$$\dot{h}\,\frac{h}{2}\Delta\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \Delta\rho(\mathbf{x}, \mathcal{X}, h) - \Delta\rho_D(\mathbf{x}). \tag{5.21}$$

Remark 13 *In the case of $\rho_D(\mathbf{x}) = 0,\ \forall \mathbf{x} \in \mathbb{R}^d$, one gets the following,*

$$\dot{h}\,\frac{h}{2}\Delta\rho(\mathbf{x}, \mathcal{X}, h) \;=\; \Delta\rho(\mathbf{x}, \mathcal{X}, h),$$

which yields the original solution of the heat equation defined by the ***heat kernel****,*

$$\dot{h} \;=\; \frac{2}{h}. \tag{5.22}$$

The solution to the ordinary differential equation (ODE) defined by (5.22) is $h(t) = \sqrt{4t + h_0^2}$. In particular, taking $h_0 = 0$ recovers the fundamental solution of the heat equation, namely the heat kernel (5.3). This observation provides a natural consistency check and highlights why Gaussian densities arise canonically in semigroup theory. Indeed, the heat kernel represents diffusion in time and is invariant under the action of its generator, the Laplacian Δ. Moreover, Remark 13 also shows that the ODE (5.22) is well-defined point-wise for every $\mathbf{x} \in \mathbb{R}^d$ independently of agent locations $\mathcal{X}$.

In case of $\rho_D(\mathbf{x}) \neq 0$, the resulting dynamics introduce nontrivial input terms into the heat equation. These inputs generally destroy point-wise consistency and prevent the evolution from being represented solely by the heat semigroup, thereby necessitating a weak (distributional) formulation of the problem. The weak formulation aligns naturally with a decentralized interpretation of the dynamics, in which computation is performed locally around agent locations rather than over the entire ambient space $\mathbb{R}^d$.

5.4.1 Using Weak Formulation

The density $\rho(\mathbf{x}, \mathcal{X}, h)$ is defined as a kernel density estimate for a finite number of agents and is therefore inherently localized around the agent positions $\{\mathbf{x}_i\}_{i=1}^N$. Naturally, rather than seeking a point-wise solution of the governing heat equation over all of $\mathbb{R}^d$, it is natural and sufficient to characterize the evolution of ρ in a weak (distributional) sense, evaluated against test functions near the agents. From this perspective, the density estimate may be interpreted as the kernel regularization of an empirical measure consisting of a finite sum of Dirac delta distributions located at the agent positions. The associated diffusion process can therefore be viewed as evolving, in time, the configuration of these Dirac masses through convolution with the heat kernel (see Appendix C), such that the resulting smooth density best approximates a prescribed target density $\rho_D(\mathbf{x})$. Again for $\dot{\mathbf{x}}_i = 0\ \forall i$, multiplying both sides of (5.21) by a test function $\phi(\mathbf{x})$ and integrating over $\mathbb{R}^d$ yields the following weak formulation:

$$\dot{h}\,\frac{h}{2}\int_{\mathbb{R}^d} \Delta\rho(\mathbf{x}, \mathcal{X}, h)\,\phi(\mathbf{x})\,d\mathbf{x} \;=\; \int_{\mathbb{R}^d} \Delta\rho(\mathbf{x}, \mathcal{X}, h)\,\phi(\mathbf{x})\,d\mathbf{x} \;-\; \int_{\mathbb{R}^d} \Delta\rho_D(\mathbf{x})\,\phi(\mathbf{x})\,d\mathbf{x}. \tag{5.23}$$

This weak formulation unlocks several important advantages. It relaxes the regularity requirements on ρ_D, allowing the dynamics to be defined even when point-wise Laplacians fail to exist. Solutions of the weak formulation inherit additional regularity through the smoothing properties of the semigroup and recover classical solutions whenever sufficient regularity exists. Finally, the weak form is inherently suited for localized and decentralized computation, since the evolution is tested only locally rather than enforced globally over $\mathbb{R}^d$.

Although test functions are typically chosen from $C_c^\infty(\mathbb{R}^d)$ in classical weak formulations (e.g., finite element methods), the requirement of compact support is not essential in the present setting. Since the density $\rho(\mathbf{x}, \mathcal{X}, h)$ is defined as a sum of rapidly decaying Gaussian-based kernels and the target density $\rho_D \in L^1(\mathbb{R}^d)$, all pairings appearing in the weak formulation are well-defined for a broader class of smooth test functions with suitable decay at infinity. In particular, test functions drawn from *Schwartz* space, i.e., $\phi \in \mathbb{S}(\mathbb{R}^d)$ can be employed without ambiguity, with differential operators acting on ρ_D in the distributional sense. In this setting, decay of the test functions replaces compact support in ensuring the validity of the integration-by-parts identities (see proof of Lemma 9). This flexibility allows the weak formulation to better reflect the global structure of the diffusion process while remaining fully consistent with classical distributional theory. Testing the weak formulation against the current density $\rho(\mathbf{x}, \mathcal{X}, h)$ yields the following relation,

$$
\begin{aligned}
\dot{h}\,\frac{h}{2}\int_{\mathbb{R}^d} \Delta\rho(\mathbf{x},\mathcal{X},h)\,\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \;&=\; \int_{\mathbb{R}^d} \Delta\rho(\mathbf{x},\mathcal{X},h)\,\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \;-\; \int_{\mathbb{R}^d} \rho_D(\mathbf{x})\,\Delta\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \\
&=\; \int_{\mathbb{R}^d} \Delta\rho(\mathbf{x},\mathcal{X},h)\big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)\,d\mathbf{x} \\
&=\; \frac{2}{h}\int_{\mathbb{R}^d} \frac{\partial}{\partial h}\rho(\mathbf{x},\mathcal{X},h)\big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)\,d\mathbf{x} \\
&=\; \frac{1}{h}\frac{\partial}{\partial h}\left(\int_{\mathbb{R}^d} \big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)^2\,d\mathbf{x}\right).
\end{aligned}
$$

The following observation provides additional insight into the left-hand side term by revealing its intrinsic non-positivity and therefore characterizes a monotone decay of density roughness under bandwidth diffusion.

Observation 2 *Note the following relations,*

$$
\begin{aligned}
\int_{\mathbb{R}^d} \Delta\rho(\mathbf{x},\mathcal{X},h)\,\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \;&=\; -\int_{\mathbb{R}^d} (\nabla\rho(\mathbf{x},\mathcal{X},h))^{\mathsf{T}}\,\nabla\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \\
&=\; -\int_{\mathbb{R}^d} \|\nabla\rho(\mathbf{x},\mathcal{X},h))\|^2\,d\mathbf{x} \;\le\; 0
\end{aligned}
$$

therefore, using the Observation 1 yields,

$$
\begin{aligned}
\frac{h}{2}\int_{\mathbb{R}^d} \Delta\rho(\mathbf{x},\mathcal{X},h)\,\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \;&=\; \int_{\mathbb{R}^d} \frac{\partial}{\partial h}\rho(\mathbf{x},\mathcal{X},h)\,\rho(\mathbf{x},\mathcal{X},h)\,d\mathbf{x} \\
&=\; \frac{1}{2}\frac{\partial}{\partial h}\left(\int_{\mathbb{R}^d} \big(\rho(\mathbf{x},\mathcal{X},h)\big)^2\,d\mathbf{x}\right) \;\le\; 0.
\end{aligned}
$$

Using Observation 2 one gets the following relation for the h state propagation equation for the case of $\dot{\mathbf{x}}_i = 0 \ \forall i$,

$$\dot{h} \;=\; \frac{2}{h}\,\frac{\dfrac{\partial}{\partial h}\left\{\displaystyle\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)^2\,d\mathbf{x}\right\}}{\dfrac{\partial}{\partial h}\left\{\displaystyle\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)\big)^2\,d\mathbf{x}\right\}} \tag{5.24}$$

Note that it is trivial to recover Remark 13 by setting $\rho_D(\mathbf{x}) = 0, \ \forall \mathbf{x} \in \mathbb{R}^d$ in (5.24). Also, the adaptive bandwidth dynamics given by (5.24) drive the system toward stationary points of the density mismatch with respect to bandwidth.

Remark 14 *The denominator in (5.24) is non-positive by Observation 2. Moreover, for Gaussian-based kernel densities,*

$$\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)\big)^2\,d\mathbf{x}\right\} \to 0$$

only asymptotically as $h \to \infty$, corresponding to the fully diffused limit where the density becomes spatially flattened. Therefore, the singularity of (5.24) is associated with the asymptotic diffusion regime rather than finite-h behavior.

The relation given by (5.24) is defined for global h value for the entire swarm under the stationary agent locations assumptions. When the agent locations evolve according to the control law (5.19), the swarm density becomes explicitly time-dependent through the trajectories $\mathbf{x}_i(t)$. Therefore, the evolution equation must additionally account for the transport induced by the motion of the agents as follows:

$$\dot{h}\,\frac{\partial}{\partial h}\rho(\mathbf{x},\mathcal{X},h) \;+\; \sum_{j=1}^{N} \nabla_{\mathbf{x}_j}^{\mathsf{T}} \rho(\mathbf{x},\mathcal{X},h)\,\dot{\mathbf{x}}_j \;=\; \Delta\rho(\mathbf{x},\mathcal{X},h) - \Delta\rho_D(\mathbf{x}). \tag{5.25}$$

Taking the weak formulation with test function $\rho(\mathbf{x}, \mathcal{X}, h)$ yields,

$$\dot{h} \int_{\mathbb{R}^d} \frac{\partial}{\partial h}\rho(\mathbf{x}, \mathcal{X}, h)\, \rho(\mathbf{x}, \mathcal{X}, h)\, d\mathbf{x} \;+\; \sum_{j=1}^{N} \int_{\mathbb{R}^d} \rho(\mathbf{x}, \mathcal{X}, h) \nabla_{\mathbf{x}_j}^{\mathsf{T}} \rho(\mathbf{x}, \mathcal{X}, h)\, d\mathbf{x}\, \dot{\mathbf{x}}_j$$
$$= \int_{\mathbb{R}^d} \rho(\mathbf{x}, \mathcal{X}, h)\big(\Delta\rho(\mathbf{x}, \mathcal{X}, h) - \Delta\rho_D(\mathbf{x})\big)\, d\mathbf{x},$$

which can be written as,

$$\dot{h}\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h)\big)^2 d\mathbf{x}\right\} \;+\; \sum_{j=1}^{N} \nabla_{\mathbf{x}_j}^{\mathsf{T}} \left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h)\big)^2 d\mathbf{x}\right\} \dot{\mathbf{x}}_j$$
$$= \frac{2}{h}\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h) - \rho_D(\mathbf{x})\big)^2 d\mathbf{x}\right\}$$

that gives the following extension to (5.24),

$$\dot{h} \;=\; \frac{2}{h}\, \frac{\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h) - \rho_D(\mathbf{x})\big)^2 d\mathbf{x}\right\}}{\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h)\big)^2 d\mathbf{x}\right\}} \;-\; \frac{\sum_{i=1}^{N} \nabla_{\mathbf{x}_i}^{\mathsf{T}} \left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h)\big)^2 d\mathbf{x}\right\} \dot{\mathbf{x}}_i}{\frac{\partial}{\partial h}\left\{\int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h)\big)^2 d\mathbf{x}\right\}} \tag{5.26}$$

Although the propagation law in the Equation (5.26) is written in a compact global form, a *decentralized* implementation requires a more careful examination of the information structure induced by each term. In particular, the evolution of the smoothing parameter depends on quantities involving pairwise interactions across the entire swarm. Even though several components of the dynamics can be locally measured by individual agents, the resulting exact propagation law requires access to globally aggregated information. To expose this structure explicitly, the denominator and numerator terms are decomposed into averages of locally measurable quantities. This decomposition reveals the precise information that must either be communicated or estimated through distributed coordination mechanisms. More specifically, each agent can evaluate contributions generated by its own local neighborhood interactions, while the complete propagation law requires the network-wide averages of these local measurements. Taking a closer look at the denominator term in the Equation (5.26),

$$
\begin{aligned}
\frac{\partial}{\partial h}\left\{\frac{1}{2}\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)\big)^2 d\mathbf{x}\right\} &= \frac{1}{2}\frac{\partial}{\partial h}\left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\big(K_h * K_h\big)\big(\mathbf{x}_i-\mathbf{x}_j\big)\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial h}\left(\frac{1}{2N}\sum_{j=1}^{N}\big(K_h * K_h\big)\big(\mathbf{x}_i-\mathbf{x}_j\big)\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\lambda_i(\mathcal{X},h) \;=\; \Lambda(\mathcal{X},h).
\end{aligned}
$$

The term $\lambda_i(\mathcal{X},h)$ is *locally measured* by agent i. However the denominator term requires the mean of all these local measurements. Hence, the propagation equation for state h requires *global information*. Similarly,

$$
\begin{aligned}
\frac{\partial}{\partial h}\left\{\frac{1}{2}\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)^2 d\mathbf{x}\right\} &= \frac{1}{2}\frac{\partial}{\partial h}\Bigg\{\left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\big(K_h * K_h\big)\big(\mathbf{x}_i-\mathbf{x}_j\big)\right) \\
&\quad - 2\left(\frac{1}{N}\sum_{i=1}^{N}\big(K_h * \rho_D\big)\big(\mathbf{x}_i\big)\right) + \int_{\mathbb{R}^d}\rho_D(\mathbf{x})^2\,\mathrm{d}\mathbf{x}\Bigg\} \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\lambda_i(\mathcal{X},h) - \frac{\partial}{\partial h}\big(K_h * \rho_D\big)\big(\mathbf{x}_i\big)\right).
\end{aligned}
$$

Combining the diffusion mismatch contribution with the transport correction term yields the following aggregated numerator quantity,

$$
\begin{aligned}
\Xi(\mathcal{X},h) &= \frac{2}{h}\frac{\partial}{\partial h}\left\{\frac{1}{2}\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)-\rho_D(\mathbf{x})\big)^2 d\mathbf{x}\right\} - \sum_{i=1}^{N}\nabla_{\mathbf{x}_i}^{\mathsf{T}}\left\{\frac{1}{2}\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h)\big)^2 d\mathbf{x}\right\}\dot{\mathbf{x}}_i \\
&= \frac{1}{N}\sum_{i=1}^{N}\frac{2}{h}\left(\lambda_i(\mathcal{X},h) - \frac{\partial}{\partial h}\big(K_h * \rho_D\big)\big(\mathbf{x}_i\big)\right) - \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{N}\sum_{j\neq i}^{N}\nabla_{\mathbf{x}_i}^{\mathsf{T}}\big(K_h * K_h\big)\big(\mathbf{x}_i-\mathbf{x}_j\big)\right)\dot{\mathbf{x}}_i \\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\frac{2}{h}\left(\lambda_i(\mathcal{X},h) - \frac{\partial}{\partial h}\big(K_h * \rho_D\big)\big(\mathbf{x}_i\big)\right) - \mu_i^{\mathsf{T}}(\mathcal{X},h)\,\dot{\mathbf{x}}_i\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\xi_i(\mathcal{X},h),
\end{aligned}
$$

which reduces the equation (5.26) into the following form:

$$\dot{h} \;=\; \frac{\Xi(\mathcal{X},h)}{\Lambda(\mathcal{X},h)} \;=\; \frac{\dfrac{1}{N}\displaystyle\sum_{j=1}^{N}\xi_j(\mathcal{X},h)}{\dfrac{1}{N}\displaystyle\sum_{j=1}^{N}\lambda_j(\mathcal{X},h)}. \tag{5.27}$$

The propagation law (5.27) describes the evolution of a single globally shared smoothing parameter. However, as demonstrated by the heat system (5.5), each agent maintains its own local smoothing state h_i, resulting in a distributed collection of density representations parameterized by $\mathcal{H} = [h_1, \ldots, h_N]^\mathsf{T}$. Therefore, the transport-induced heat equation given by (5.25) must be generalized from a single diffusion process to a coupled system of interacting heat equations whose synchronization is enforced through additional consensus coupling terms as follows,

$$\dot{h}_i \frac{\partial}{\partial h_i}\rho(\mathbf{x},\mathcal{X},h_i) \;+\; \sum_{j=1}^{N}\nabla_{\mathbf{x}_j}^\mathsf{T}\rho(\mathbf{x},\mathcal{X},h_i)\,\dot{\mathbf{x}}_j \;=\; \Delta\rho(\mathbf{x},\mathcal{X},h_i) - \Delta\rho_D(\mathbf{x}) \;+\; \mathcal{C}_i(\mathbf{x},\mathcal{X},\mathcal{H}),$$

for all $i \in \{1,\ldots,N\}$, where,

$$\mathcal{C}_i(\mathbf{x},\mathcal{X},\mathcal{H}) \;=\; -\beta\sum_{j\neq i} a_{ij}\big(\rho(\mathbf{x},\mathcal{X},h_i) - \rho(\mathbf{x},\mathcal{X},h_j)\big).$$

Note that consensus term is defined over densities ρ, as oppose to density errors ϑ defined in [5.16]. However, $\vartheta_i - \vartheta_j = \rho_i - \rho_j$ in the case of constant and universal desired density ρ_D. The corresponding weak formulation generated by the consensus coupling becomes,

$$\int_{\mathbb{R}^d}\mathcal{C}_i(\mathbf{x},\mathcal{X},\mathcal{H})\,\rho(\mathbf{x},\mathcal{X},h_i)\,d\mathbf{x} \;=\; -\beta\sum_{i\neq j}a_{ij}\int_{\mathbb{R}^d}\big(\rho(\mathbf{x},\mathcal{X},h_i) - \rho(\mathbf{x},\mathcal{X},h_j)\big)\rho(\mathbf{x},\mathcal{X},h_i)\,d\mathbf{x}.$$

Finally, using this consensus term in (5.27) yields the following individual h_i dynamics,

$$\dot{h}_i = \frac{\frac{1}{N}\sum_{j=1}^{N} \xi_j(\mathcal{X}, h_i)}{\frac{1}{N}\sum_{j=1}^{N} \lambda_j(\mathcal{X}, h_i)} + \frac{\int_{\mathbb{R}^d} \mathcal{C}_i(\mathbf{x}, \mathcal{X}, \mathcal{H})\, \rho(\mathbf{x}, \mathcal{X}, h_i)\, d\mathbf{x}}{\frac{1}{N}\sum_{j=1}^{N} \lambda_j(\mathcal{X}, h_i)} \tag{5.28}$$

5.4.2 Decentralized Implementation

Although the coupled propagation law given by (5.28) provides a continuum-consistent synchronization mechanism between local density representations, its direct implementation introduces severe computational and structural challenges in a decentralized setting.

First, the weak formulation generated by the continuous density consensus coupling,

$$\int_{\mathbb{R}^d} \mathcal{C}_i(\mathbf{x}, \mathcal{X}, \mathcal{H})\, \rho(\mathbf{x}, \mathcal{X}, h_i)\, d\mathbf{x} = -\beta \sum_{j \neq i} a_{ij} \int_{\mathbb{R}^d} \big(\rho(\mathbf{x}, \mathcal{X}, h_i) - \rho(\mathbf{x}, \mathcal{X}, h_j)\big) \rho(\mathbf{x}, \mathcal{X}, h_i)\, d\mathbf{x},$$

introduces prohibitive computational complexity compared to the original heat propagation dynamics. Because each local density field is already expressed as a kernel expansion over the finite swarm configuration $\mathcal{X}$, evaluating pairwise synchronization integrals over these full density fields requires calculating complicated interactions between all particles simultaneously. This yields an effective triple interaction structure whose computational cost scales between $\mathcal{O}(N^2)$ and $\mathcal{O}(N^3)$ per agent, depending on evaluation reuse. However, since all local density representations are evaluated using the exact same underlying spatial configuration $\mathcal{X}$, the dominant source of disagreement between neighboring agents stems primarily from discrepancies between their individual smoothing parameters h_i. Therefore, rather than synchronizing the full high-dimensional density fields directly, one can introduce a structurally reduced synchronization mechanism directly on the bandwidth states. The localized bandwidth coupling scales linearly with neighborhood size and simplifies individual propagation dynamics to:

$$\dot{h}_i = \frac{\Xi_i}{\Lambda_i} - k_h \sum_{j \neq i}^{N} a_{ij}(h_i - h_j), \tag{5.29}$$

where $a_{ij} = K_{h_i}(\mathbf{x}_i - \mathbf{x}_j)$, and $k_h > 0$ is a consensus gain enforcing alignment of local spatial scales.

Second, evaluating the exact propagation law derived from the weak formulation (5.28) requires the following agent-dependent quantities,

$$\Xi_i = \frac{1}{N} \sum_{j=1}^{N} \xi_j(\mathcal{X}, h_i),$$
$$\Lambda_i = \frac{1}{N} \sum_{j=1}^{N} \lambda_j(\mathcal{X}, h_i).$$

Unlike conventional distributed averaging problems, the local contributions appearing in these averages are evaluated using the receiving agent's bandwidth state h_i rather than the contributing agent's local bandwidth state h_j. Therefore, the quantities Ξ_i and Λ_i do not correspond to standard network averages of locally measurable signals and cannot be recovered directly through conventional dynamic average consensus mechanisms. To obtain a scalable decentralized implementation, we instead introduce the approximations

$$\Xi_i \approx \Xi(\mathcal{X}, \mathcal{H}) = \frac{1}{N} \sum_{j=1}^{N} \xi_j(\mathcal{X}, h_j),$$
$$\Lambda_i \approx \Lambda(\mathcal{X}, \mathcal{H}) = \frac{1}{N} \sum_{j=1}^{N} \lambda_j(\mathcal{X}, h_j),$$

which replace the agent-dependent evaluations with averages of quantities that are locally computable by each contributing agent. This approximation becomes exact on the bandwidth-consensus subspace $h_1 = \cdots = h_N$, and its accuracy improves as the bandwidth synchronization term in (5.29) drives neighboring bandwidth states toward agreement. It is important to note that, the bandwidth consensus dynamics serve a dual purpose: they not only synchronize neighboring smoothing scales but also reduce the approximation error introduced by replacing the agent-dependent quantities Ξ_i and Λ_i with globally averaged quantities compatible with dynamic average consensus estimation. Because these quantities rely on globally aggregated information, they are not directly accessible to agents operating under localized communication topologies. Furthermore, because $\Lambda(\mathcal{X}, \mathcal{H})$ acts as a time-varying denominator, recovering it via distributed consensus estimators introduces structural vulnerabilities: transient estimator tracking errors or initialization lag can drive the local denominator estimate near zero, triggering numerical instabilities in $\dot{h}_i$.

To bypass these estimation and singularity challenges, the following two distinct structural remedies to approximate the continuous dynamics are investigated.

Remedy 1: Directional Gradient Simplification ($\Lambda_i \to -1$)

By Remark 14, the global denominator $\Lambda(\mathcal{X}, \mathcal{H})$ is strictly negative-definite away from the fully diffused limit ($h \to \infty$). Thus, one alternative is to completely eliminate the tracking of the denominator scalar and substitute a constant value of -1. Under this simplification, the bandwidth dynamics in (5.29) reduce to:

$$\dot{h}_i \;=\; -\widehat{\Xi}_i - k_h \sum_{j \neq i}^{N} a_{ij}(h_i - h_j).$$

This drastically minimizes communication overhead and network state complexity. Because agents no longer need to track $\Lambda(\mathcal{X}, \mathcal{H})$, the required distributed dynamic consensus estimators and their associated internal states are cut exactly in half. However, the explicit connection to the physical time-scale of continuous spatial diffusion is broken; the system no longer recovers the fundamental solution $h(t) = \sqrt{4t + h_0^2}$ when $\rho_D = 0$. Additionally, an explicit tuning gain must be introduced to account for different swarm dimensions.

Remedy 2: Coincident Agent Approximation of the Non-Positivity Scaling

An alternate approach leverages a high-density local approximation to isolate the spatial dependency of $\Lambda(\mathcal{X}, \mathcal{H})$. For the purpose of evaluating the denominator alone, one can evaluate the interaction under the assumption that local variations dominate, approximating the local swarm configuration as locally coincident ($\mathbf{x}_i \approx \mathbf{x}_j$, $\forall j$). Using the Gaussian self-convolution given by Remark 12, and evaluating this at zero spatial separation yields the peak amplitude in a d-dimensional space:

$$K_{h_i\sqrt{2}}(\mathbf{0}) \;=\; \frac{1}{\left(h_i\sqrt{2}\sqrt{\pi}\right)^d} \;=\; \frac{1}{\left(2\pi\right)^{d/2} h_i^d}.$$

Substituting this localized kernel interaction directly into the definition of Λ_i yields an analytical, h_i state-dependent scalar expression for the local denominator that requires zero inter-agent communication:

$$\Lambda(h_i) \;\approx\; \frac{1}{2}\frac{\partial}{\partial h_i}\left[\frac{1}{\left(2\pi\right)^{d/2} h_i^d}\right] \;=\; -\frac{d}{2\left(2\pi\right)^{d/2} h_i^{d+1}}. \tag{5.30}$$

This retains the structural dependency on the spatial dimension d and the current bandwidth h_i, preserving the natural damping behavior of the continuous diffusion model. More importantly, the mathematical singularity is isolated strictly to $h_i = 0$. However, treating the denominator as a purely local function yields a mismatch with the true continuous weak formulation when the swarm undergoes extreme spatial dispersion.

Evaluating these options, remedy 2 offers the most balanced compromise, maintaining continuum structural consistency while localizing the singularity structure to the physically meaningful limit $h_i \to 0$. Under this framework, agents only need to estimate the global numerator term $\Xi(\mathcal{X}, \mathcal{H})$. To asymptotically recover this network-wide average using only neighbor-to-neighbor communications, a distributed proportional dynamic average consensus estimator [80] can be employed. Each agent maintains an internal estimator state $[\widehat{\Xi}_i \;\; w_i]^\mathsf{T} \in$

$\mathbb{R}^2$ governed by:

$$\begin{aligned} \dot{w}_i &= -\gamma w_i - \sum_{j \neq i}^{N} a_{ij}(t)\big(\widehat{\Xi}_i - \widehat{\Xi}_j\big), \\ \widehat{\Xi}_i &= w_i + \xi_i(\mathcal{X}, h_i), \end{aligned} \tag{5.31}$$

where $\gamma \geq 0$ introduces proportional damping to attenuate accumulation tracking errors stemming from time-varying local agent measurements $\xi_i(\mathcal{X}, h_i)$. Assuming the interaction graph remains connected and the local measurements vary sufficiently slowly relative to the estimator dynamics, $\widehat{\Xi}_i$ tracks the network average $\frac{1}{N}\sum_{j=1}^{N} \xi_j(\mathcal{X}, h_j)$ with arbitrarily small error under sufficiently fast estimator dynamics. Combining the motion control laws, the localized bandwidth consensus, the analytical denominator reduction (5.30), and the dynamic numerator estimator (5.31), the closed-loop, decentralized swarm dynamics become:

$$\begin{aligned} \dot{\mathbf{x}}_i &= -\frac{\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h_i)}{\rho(\mathbf{x}_i, \mathcal{X}, h_i)}, \\ \dot{h}_i &= -C_d h_i^{d+1}\widehat{\Xi}_i - k_h \sum_{j \neq i}^{N} K_{h_i}(\mathbf{x}_i - \mathbf{x}_j)(h_i - h_j), \\ \dot{w}_i &= -\gamma w_i - \sum_{j \neq i}^{N} K_{h_i}(\mathbf{x}_i - \mathbf{x}_j)(\widehat{\Xi}_i - \widehat{\Xi}_j), \\ \widehat{\Xi}_i &= w_i + \xi_i(\mathcal{X}, h_i), \end{aligned} \tag{5.32}$$

where $C_d = \dfrac{2(2\pi)^{d/2}}{d}$. It should be emphasized that the resulting decentralized bandwidth dynamics given in (5.32) no longer coincide exactly with the centralized propagation law (5.28). During transient operation, estimator errors introduce perturbations into the bandwidth evolution, and the resulting closed-loop system should therefore be interpreted as an approximation of the ideal heat-flow dynamics. Furthermore, additional analysis is required to establish positive invariance of the admissible set $h_i > 0$ and to characterize the behavior of (5.32) near such degeneracies. These questions are left for future investigation.

5.5 Numerical Simulations

To illustrate the behavior of the decentralized implementation developed in Section 5.4.2, numerical simulations are conducted using the closed-loop dynamics (5.32). The objectives of these experiments are twofold. First, we investigate whether the decentralized approximation successfully drives the swarm toward a prescribed target density. Second, we examine how the adaptive bandwidth dynamics influence the quality of finite-agent density representations and how the resulting bandwidth values depend on swarm size. The desired density is

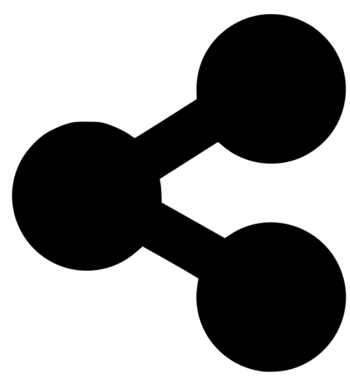

Figure 5.1: Desired Density Profile

generated from a grayscale image given by Figure 5.1 and represented through the kernel-density framework introduced previously. In all simulations, agent positions are initialized randomly over the domain and bandwidth states are initialized uniformly. The decentralized controller is then evolved until the swarm approaches its steady-state configuration.

Control Response with Different Population Sizes

Figure 5.2 illustrates the evolution of the swarm density for three different population sizes. In all cases, the decentralized controller successfully drives the swarm toward the desired density despite relying solely on localized communication and distributed bandwidth adap-

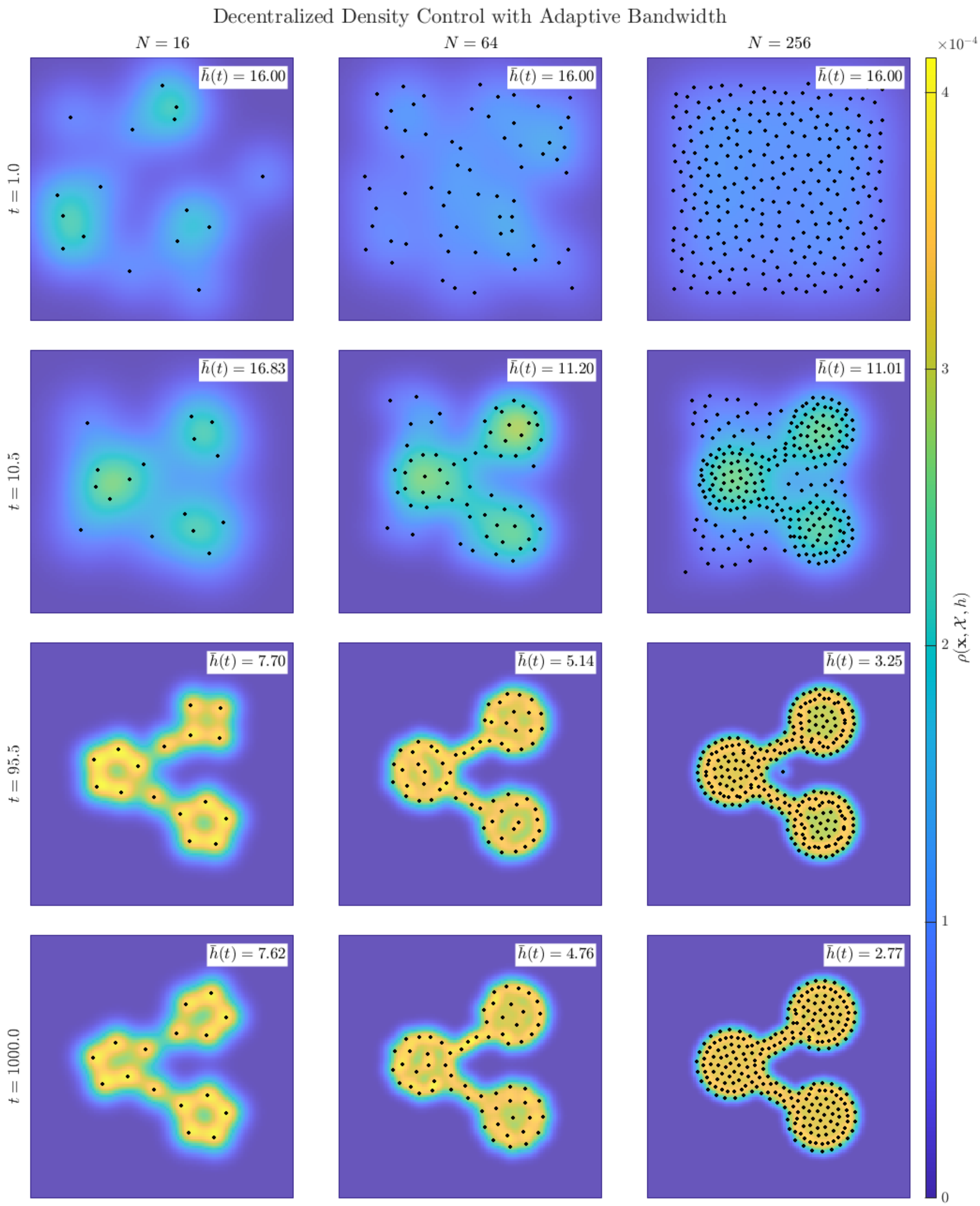

Figure 5.2: Evolution of the decentralized adaptive-bandwidth density controller for $N = 16$, 64, and 256 agents over a 100×100 domain. Each row corresponds to a different swarm size and each column corresponds to a snapshot in time. Colored regions denote the swarm density $\rho(\mathbf{x}, \mathcal{X}, \bar{h})$ and black markers indicate agent locations.

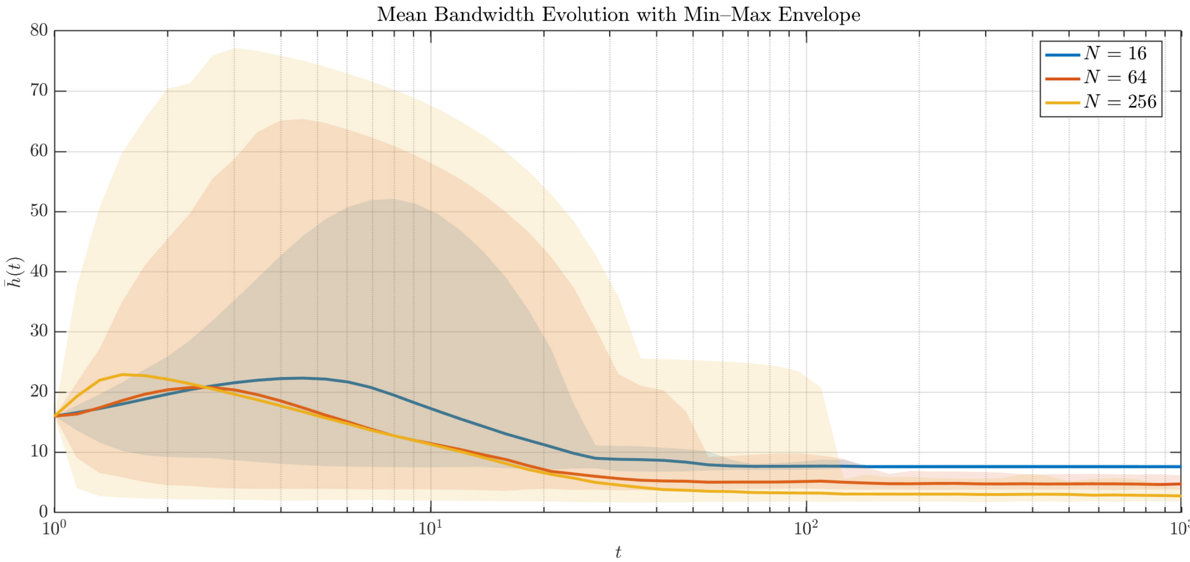

Figure 5.3: Mean bandwidth evolution for the decentralized adaptive density controller with $N = 16$, 64, and 256 agents. Solid curves show the mean bandwidth $\bar{h}(t)$, while shaded envelopes indicate the range $\min_i h_i(t) \leq h_i(t) \leq \max_i h_i(t)$.

tation. As the number of agents increases, the resulting density representation captures progressively finer geometric features of the target distribution. This behavior is expected since larger swarms provide a denser sampling of the desired density and therefore require less artificial smoothing to construct an accurate kernel-density estimate.

A notable feature of the proposed controller is that the bandwidth states are not prescribed a priori but instead evolve according to the decentralized adaptation law. Therefore, the amount of smoothing required by the density representation emerges directly from the closed-loop dynamics.

The corresponding bandwidth trajectories are shown in Figure 5.3. For all swarm sizes, the adaptive law initially increases the bandwidth, thereby enlarging the kernel support and smoothing large-scale discrepancies between the current and desired densities. As the swarm reorganizes and approaches the target distribution, the bandwidth gradually decreases, allowing increasingly finer spatial features to be represented. More importantly, the steady-state bandwidth exhibits a clear dependence on swarm size. Larger populations converge to smaller equilibrium bandwidth values, whereas smaller populations require larger amounts of smoothing to compensate for the limited spatial sampling provided by the finite number of agents. This behavior is consistent with the asymptotic relationship established in (3.13) and may be summarized qualitatively as

$$N \uparrow \implies h^\star \downarrow .$$

The simulations therefore suggest that the proposed adaptation law automatically learns the amount of smoothing required by the finite-agent representation. Rather than selecting a fixed bandwidth manually, the controller continuously adjusts the kernel scale to balance density accuracy and spatial resolution throughout the evolution. As the swarm size increases, the adaptive mechanism naturally reduces the required smoothing, enabling higher-fidelity approximations of the desired density while preserving the decentralized structure of the algorithm.

Comparison with Fixed-Bandwidth Density Representations

While Figures 5.2 and 5.3 demonstrate that the proposed adaptation law automatically selects bandwidth values according to the available swarm population, they do not directly quantify the benefit of adapting the bandwidth during execution. To assess the contribution of the bandwidth dynamics, additional simulations are performed using a fixed population size of $N = 32$ agents. Four controllers are considered. Three simulations employ constant bandwidth values $h \in \{6.0, 7.5, 9.0\}$ throughout the entire evolution, while the fourth uses the adaptive bandwidth dynamics proposed in (5.32). All simulations are initialized from identical random agent configurations and are tasked with reproducing the same desired density shown in Figure 5.1.

Figure 5.4 compares the resulting steady-state density representations. Although all controllers drive the swarm toward the desired distribution, the quality of the final approxi-

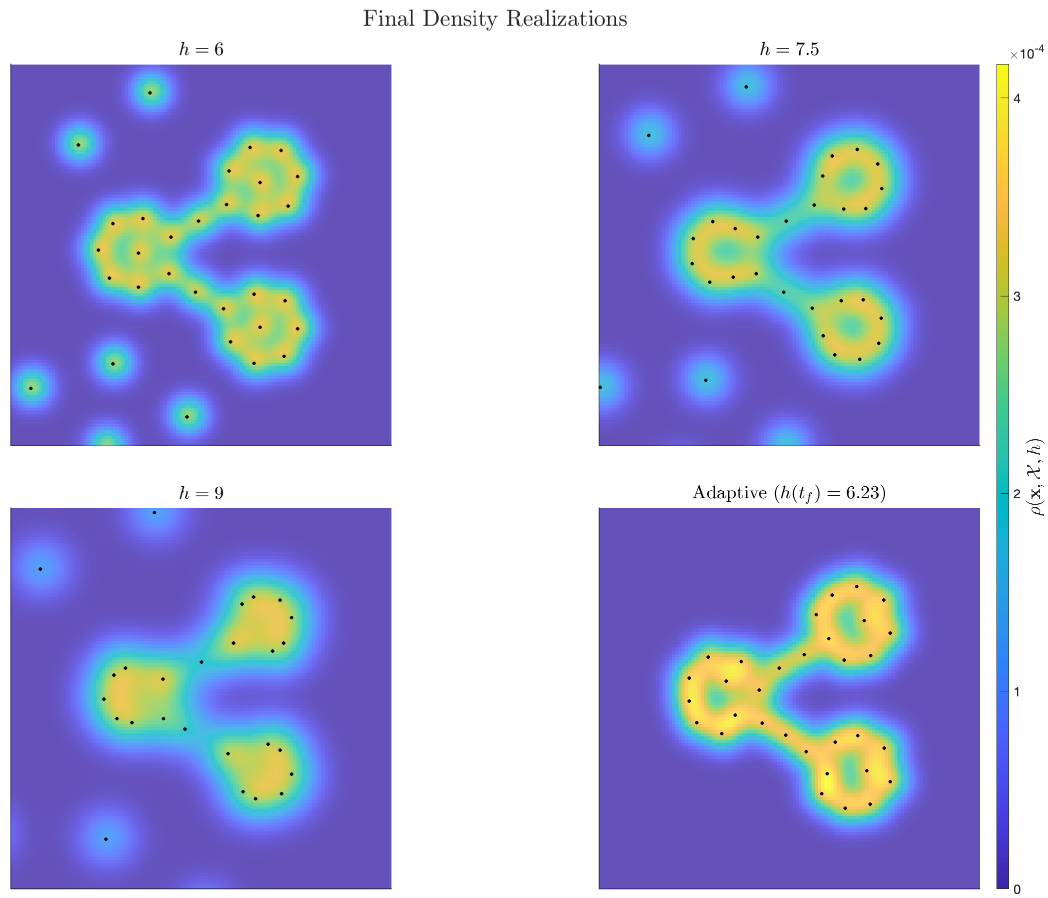

Figure 5.4: Comparison of steady-state density representations obtained using fixed and adaptive bandwidths for a swarm of $N = 32$ agents. The fixed-bandwidth controllers employ constant kernel scales $h = 6$, $h = 7.5$, and $h = 9$, while the adaptive controller evolves the bandwidth according to (5.32) and converges to $\bar{h} \approx 6.23$.

mation varies significantly with the selected bandwidth value. Smaller bandwidths preserve fine geometric features but may suffer from insufficient smoothing during transient evolution, whereas larger bandwidths provide stronger regularization at the expense of reduced spatial resolution. Because the optimal amount of smoothing depends on the current state of the swarm, no single constant bandwidth performs uniformly well throughout the entire trajectory. The adaptive controller avoids this trade-off by continuously adjusting the kernel scale in response to the evolving density mismatch. Starting from a larger initial bandwidth, the adaptation law first increases the effective smoothing level to facilitate large-scale density reorganization and subsequently reduces the bandwidth as the swarm approaches the desired distribution. In the present simulation, the mean bandwidth converges to approximately $h^\star \approx 6.23$, indicating that the controller automatically selects a smoothing level consistent with the finite-agent representation required at equilibrium.

Figure 5.5 provides a quantitative comparison through the density-error functional $\Phi(\mathcal{X}(t), \bar{h}(t))$

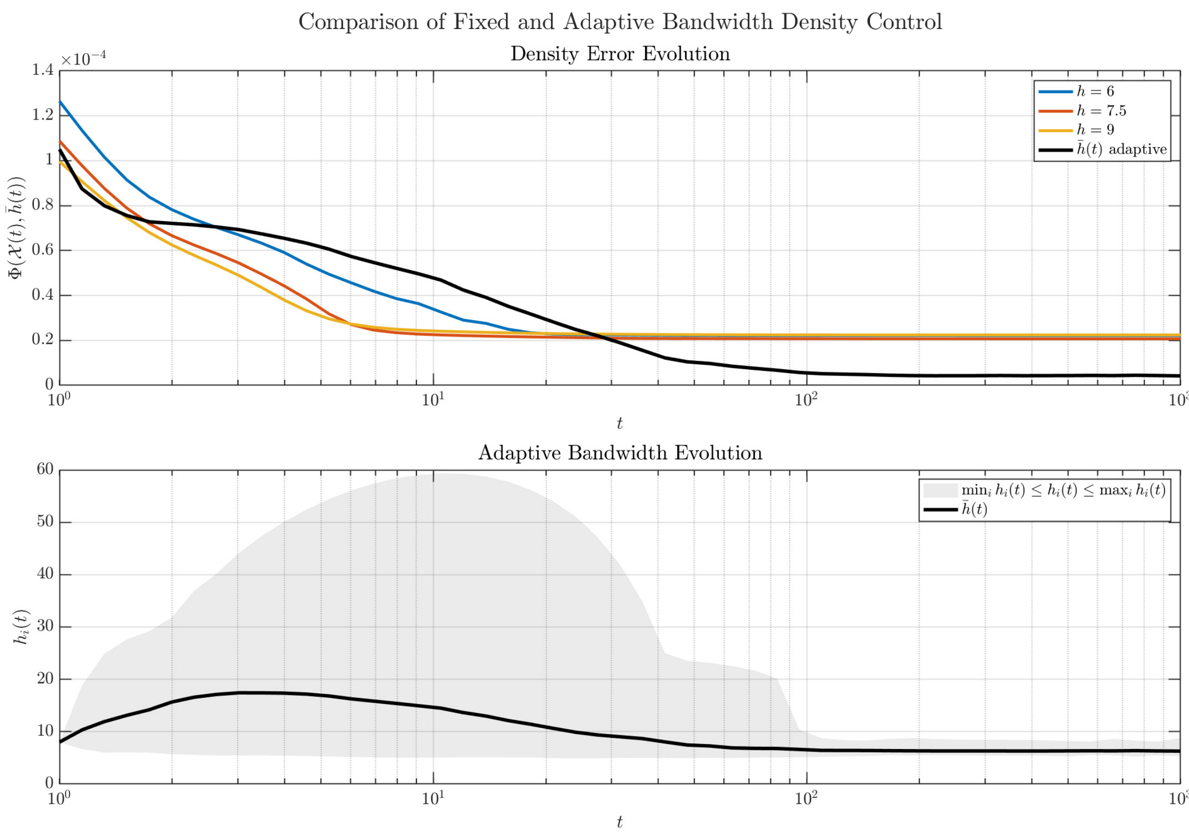

Figure 5.5: Performance comparison between fixed and adaptive bandwidth density controllers for $N = 32$ agents. The upper panel shows the density-error functional $\Phi(\mathcal{X}(t), \bar{h}(t))$ for three fixed bandwidth values and the proposed adaptive bandwidth strategy. The lower panel shows the corresponding adaptive bandwidth evolution, where the solid curve denotes the mean bandwidth $\bar{h}(t)$ and the shaded region indicates the range $\min_i h_i(t) \le h_i(t) \le \max_i h_i(t)$.

defined by (5.16). Among all tested configurations, the adaptive controller achieves the lowest final density error despite starting from a larger initial bandwidth. Moreover, the accompanying bandwidth trajectory reveals how the controller transitions between coarse and fine density representations during the evolution. This indicates that a single fixed bandwidth cannot simultaneously provide the smoothing required during early transient evolution and the spatial resolution required near equilibrium. By continuously adjusting the kernel scale, the adaptive controller is able to satisfy both objectives via improving both transient and steady-state density reconstruction performance within a single closed-loop framework.

Collectively, the simulations demonstrate that the decentralized approximation introduced in Section 5.4.2 preserves the essential behavior of the continuous density-feedback formulation while remaining implementable using only local communication. Furthermore, the adaptive bandwidth dynamics automatically compensate for finite-agent effects, allowing the swarm to balance smoothing and spatial resolution without requiring manual bandwidth selection.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Concluding Remarks

This dissertation investigated the problem of guiding and controlling large-scale robotic swarms through density-based representations and decentralized feedback mechanisms. By shifting the focus from individual agent trajectories to collective density evolution, the proposed framework enables scalable, robust, and interpretable swarm coordination strategies that are well suited for large populations operating under local information.

At the guidance level, a probabilistic density guidance framework was developed to encode desired swarm behaviors in terms of spatial density distributions. A constrained Markov chain synthesis approach was presented to generate admissible density evolution sequences that respect motion, safety, and flow constraints, providing a principled mechanism for high-level swarm coordination. This formulation enables systematic integration of environmental structure and mission requirements directly into the guidance layer.

At the control level, decentralized density feedback laws inspired by the heat equation were proposed to enable individual agents to track desired density distributions using only local information. By leveraging nonparametric local density estimation, the resulting control laws are fully decentralized and scalable. Rigorous analysis established stability and convergence properties in the continuum limit, providing formal guarantees for large-population behavior.

Safety considerations were addressed through a detailed analysis of collision avoidance properties of density-based control laws. A modified feedback strategy was analyzed, and sufficient conditions were derived under which inter-agent collisions are avoided. Explicit lower bounds on the achievable minimum inter-agent distance were established for a class of kernel functions, offering theoretical insight into the interplay between density smoothing and collision avoidance.

Finally, the dissertation examined adaptive local density estimation mechanisms for decentralized swarm control. Fundamental relationships between weak formulations of partial differential equations, the heat operator, and Gaussian kernel based density estimators were established. Dynamic average consensus based estimation schemes were developed to improve density reconstruction and promote well-distributed swarm configurations in the absence of global information.

Together, these contributions form a cohesive hierarchical framework that bridges high-level probabilistic guidance, decentralized density feedback control, safety analysis, and adaptive estimation. The results demonstrate that density-based approaches provide a powerful and flexible foundation for scalable swarm guidance and control with provable guarantees.

6.2 Further Directions

Several promising directions remain for extending the results of this dissertation. With respect to decentralized density feedback control, one natural extension is the incorporation of stochastic effects in agent dynamics and sensing. While the present work primarily considers deterministic agent motion, real-world swarm systems are subject to process noise, sensor errors, and actuation uncertainty. Incorporating stochasticity through Fokker-Planck type formulations or stochastic density evolution models could provide more realistic performance guarantees.

Another direction concerns asynchronous updates and communication delays among agents. The proposed control and estimation schemes assume synchronized updates, whereas practical systems often operate asynchronously. Extending the analysis to asynchronous or event-triggered implementations would further enhance robustness and applicability. Additionally, the framework could be generalized to accommodate broader classes of agent dynamics beyond the models considered in this dissertation. Investigating density feedback laws for systems with heterogeneous dynamics or higher-order motion models remains an open area of research. Finally, relaxing the assumption that desired density distributions are directly available at agent locations, i.e., allowing specifications at arbitrary spatial locations or through indirect measurements, would increase flexibility in practical deployments.

From the perspective of collision avoidance, several avenues warrant further investigation. While sufficient conditions were derived to guarantee collision avoidance under modified density-based feedback laws, extending these results to fully cover heat-equation driven dynamics remains an open challenge. In particular, tighter analytical bounds on the minimum achievable inter-agent distance, especially in higher-dimensional spaces, would provide sharper safety guarantees. Moreover, the present analysis assumes fixed and homogeneous smoothing parameters in the density estimation process. Studying collision avoidance properties under variable or spatially local smoothing parameters could yield less conservative bounds and improved performance, particularly in non-uniform density scenarios.

With respect to adaptive density estimation, future work includes the development of formal convergence proofs for the modified consensus and gradient-tracking mechanisms introduced in this dissertation. While empirical and analytical evidence suggests improved density reconstruction, a complete theoretical characterization would strengthen the foundations of the proposed estimation framework. Further investigation is also needed into the role of communication topology and connectivity graphs in defining local density estimates. Different graph structures may significantly affect estimation accuracy and convergence rates. In addition, the impact of imperfect onboard convolution operations for estimating desired density distributions, particularly under limited sensing or computational constraints, remains an important practical consideration. Finally, while the heat equation provides a natural and analytically tractable mechanism for deriving adaptive smoothing parameters, it represents only one possible design choice. Alternative control laws that prioritize constraint satisfaction, optimality with respect to different cost metrics, or task-specific objectives may offer improved performance in certain applications and constitute an interesting direction for future research.

BIBLIOGRAPHY

[1] Arggonauts (karlsruhe, germany). `https://arggonauts.de`. Accessed: 2018-03-23.

[2] Bangalorerobotics (bangalore, india). `http://bangalorerobotics.in/#close`. Accessed: 2018-03-23.

[3] Cfis (arnex-sur-nyon, switzerland). `https://oceandiscovery.xprize.org/teams/cfis`. Accessed: 2018-03-23.

[4] Shell ocean discovery xprize. `https://oceandiscovery.xprize.org`. Accessed: 2018-03-23.

[5] Team tao (newcastle, united kingdom). `http://www.team-tao.org`. Accessed: 2018-03-23.

[6] B. Açıkmeşe and D. S. Bayard. A markov chain approach to probabilistic swarm guidance. *American Control Conference, Montreal, Canada*, pages 6300–6307, 2012.

[7] B. Açıkmeşe and D. S. Bayard. Markov chain approach to probabilistic guidance for swarms of autonomous agents. *In Press, Asian Journal of Control*, 2015.

[8] B. Açıkmeşe and D.S. Bayard. Probabilistic swarm guidance for collaborative autonomous agents. *Revised version in review for IEEE Trans. on Automatic Control*, 2013.

[9] B. Açıkmeşe, N. Demirer, B. Açıkmeşe, and M. Harris. Convex necessary and sufficient conditions for density safety constraints in markov chain synthesis. *Accepted to IEEE Trans. on Automatic Control*, 2015.

[10] Behcet Acikmese and David S Bayard. Probabilistic swarm guidance for collaborative autonomous agents. In *American Control Conference (ACC), 2014*, pages 477–482. IEEE, 2014.

[11] A. Arapostathis, R. Kumar, and S. Tangirala. Controlled markov chains with safety upper bound. *IEEE Transactions on Automatic Control*, 48(7):1230–1234, 2003.

[12] Saptarshi Bandyopadhyay, Soon-Jo Chung, and Fred Y Hadaegh. Probabilistic swarm guidance using optimal transport. In *2014 IEEE Conference on Control Applications (CCA)*, pages 498–505. IEEE, 2014.

[13] Laura Barnes, Wendy Alvis, MaryAnne Fields, Kimon Valavanis, and Wilfrido Moreno. Swarm formation control with potential fields formed by bivariate normal functions. In *Control and Automation, 2006. MED'06. 14th Mediterranean Conference on*, pages 1–7. IEEE, 2006.

[14] Scott A Basinger, David Palacios, Marco B Quadrelli, and Grover A Swartzlander. Optics of a granular imaging system (ie orbiting rainbows). In *UV/Optical/IR Space Telescopes and Instruments: Innovative Technologies and Concepts VII*, volume 9602, page 96020E. International Society for Optics and Photonics, 2015.

[15] Levent Bayindir and Erol Şahin. A review of studies in swarm robotics. *Turkish Journal of Electrical Engineering & Computer Sciences*, 15(2):115–147, 2007.

[16] Gerardo Beni. From swarm intelligence to swarm robotics. In *International Workshop on Swarm Robotics*, pages 1–9. Springer, 2004.

[17] A. Berman and R. J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences.* SIAM, 1994.

[18] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control.* Springer, 2008.

[19] S. Bochner. *Harmonic Analysis and the Theory of Probability.* California Monographs in mathematical sciences. University of California Press, 1955.

[20] Eric Bonabeau. Swarm intelligence: From natural to artificial systems. *Oxford University Press google schola*, 2:25–34, 1999.

[21] Jean-Michel Bony. Principe du maximum, inégalité de harnack et unicité du probleme de cauchy pour les opérateurs elliptiques dégénérés. *Ann. Inst. Fourier (Grenoble)*, 19(1):277–304, 1969.

[22] Georges Bouligand. *Introducion a la geometrie infinitesimale directe.* Paris: Gauthiers-Villars, 1932.

[23] S. Boyd, P. Diaconis, P. Parillo, and L. Xiao. Fastest mixing Markov Chain on graphs with symmetries. *SIAM Jrnl. of Optimization*, 20(2):792–819, 2009.

[24] S. Boyd, P. Diaconis, and L. Xiao. Fastest mixing Markov Chain on a graph. *SIAM Review*, 46:667–689, 2004.

[25] 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.

[26] Haïm Brezis. On a characterization of flow-invariant sets. *Communications on Pure and Applied Mathematics*, 23(2):261–263, 1970.

[27] Rodney Brooks. A robust layered control system for a mobile robot. *IEEE journal on robotics and automation*, 2(1):14–23, 1986.

[28] Rodney A Brooks. How to build complete creatures rather than isolated cognitive simulators. *Architectures for intelligence*, pages 225–239, 1991.

[29] Kenneth F Caluya and Abhishek Halder. Gradient flow algorithms for density propagation in stochastic systems. *IEEE Transactions on Automatic Control*, 65(10):3991–4004, 2019.

[30] Julie C Castillo-Rogez, Marco Pavone, Jeffrey A Hoffman, and Issa AD Nesnas. Expected science return of spatially-extended in-situ exploration at small solar system bodies. In *Aerospace Conference, 2012 IEEE*, pages 1–15. IEEE, 2012.

[31] Luiz Chaimowicz, Nathan Michael, and Vijay Kumar. Controlling swarms of robots using interpolated implicit functions. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2487–2492. IEEE, 2005.

[32] Airlie Chapman. Advection on graphs. In *Semi-Autonomous Networks*, pages 3–16. Springer, 2015.

[33] Airlie Chapman and Mehran Mesbahi. Semi-autonomous consensus: network measures and adaptive trees. *IEEE Transactions on Automatic Control*, 58(1):19–31, 2013.

[34] Kai Lai Chung. *A course in probability theory.* Academic press, 2001.

[35] M. Corless and A. E. Frazho. *Linear Systems and Control: An Operator Perspective.* Marcel Dekker, 2003.

[36] Di Cui and Huiping Li. Density regulation of large-scale robotic swarm using robust model predictive mean-field control. *Automatica*, 169:111832, 2024.

[37] Mathias Hudoba de Badyn, Utku Eren, Behçet Açikmeşe, and Mehran Mesbahi. Optimal mass transport and kernel density estimation for state-dependent networked dynamic systems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 1225–1230. IEEE, 2018.

[38] Mathias Hudoba De Badyn, Erik Miehling, Dylan Janak, Behçet Açıkmeşe, Mehran Mesbahi, Tamer Başar, John Lygeros, and Roy S Smith. Discrete-time linear-quadratic regulation via optimal transport. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 3060–3065. IEEE, 2021.

[39] M.C. de Oliveira, J. Bernussou, and J.C. Geromel. Anew discrete-time robust stability condition. *Systems and Control Letters*, pages 261–265, 1999.

[40] N. Demirer, B. Açıkmeşe, and M. Harris. Convex optimization formulation of density upper bound constraints in Markov chain synthesis. *American Control Conference*, pages 483– 488, 2014.

[41] N. Demirer, B. Açıkmeşe, and C. Pehlivanturk. Density control for decentralized autonomous agents with conflict avoidance. *IFAC World Congress, South Africa*, pages 11715–11721, 2014.

[42] Nazli Demirer. *Density Control of Multi-Agent Systems with Safety Constraints: A Markov Chain Approach*. PhD thesis, 2017.

[43] Nazlı Demirer and Behçet Açıkmeşe. Probabilistic density control for swarm of decentralized on-off agents with safety constraints. In *2015 American Control Conference (ACC)*, pages 5238–5244. IEEE, 2015.

[44] Nazlı Demirer, Utku Eren, and Behçet Açıkmeşe. Decentralized probabilistic density control of autonomous swarms with safety constraints. *Autonomous Robots*, 39(4):537–554, 2015.

[45] Jaydev P Desai, James P Ostrowski, and Vijay Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE transactions on Robotics and Automation*, 17(6):905–908, 2001.

[46] Franck Djeumou, Zhe Xu, and Ufuk Topcu. Probabilistic swarm guidance subject to graph temporal logic specifications. In *Robotics: Science and Systems*, 2020.

[47] Marco Dorigo, Vito Trianni, Erol Şahin, Roderich Groß, Thomas H Labella, Gianluca Baldassarre, Stefano Nolfi, Jean-Louis Deneubourg, Francesco Mondada, Dario Floreano, et al. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2):223–245, 2004.

[48] Karthik Elamvazhuthi, Chase Adams, and Spring Berman. Coverage and field estimation on bounded domains by diffusive swarms. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 2867–2874. IEEE, 2016.

[49] Karthik Elamvazhuthi and Spring Berman. Mean-field models in swarm robotics: A survey. *Bioinspiration & Biomimetics*, 15(1):015001, 2020.

[50] Karthik Elamvazhuthi and Spring Berman. Density stabilization strategies for nonholonomic agents on compact manifolds. *IEEE Transactions on Automatic Control*, 69(3):1448–1463, 2023.

[51] Klaus-Jochen Engel and Rainer Nagel. *One-parameter semigroups for linear evolution equations*. Springer, 2000.

[52] Klaus-Jochen Engel and Rainer Nagel. *A short course on operator semigroups*. Springer, 2006.

[53] U. Eren and B. Açıkmeşe. Velocity field generation for density control of swarms using heat equation and smoothing kernels. *IFAC World Congress, Toulouse, France*, 2017.

[54] Lawrence C Evans. *Partial differential equations*, volume 19. American mathematical society, 2022.

[55] M. Fiedler. *Special Matrices and Their Applications in Numerical Mathematics*. Dover, 2008.

[56] Greg Foderaro, Silvia Ferrari, and Thomas A Wettergren. Distributed optimal control for multi-agent trajectory optimization. *Automatica*, 50(1):149–154, 2014.

[57] Terrence Fong, Illah Nourbakhsh, and Kerstin Dautenhahn. A survey of socially interactive robots. *Robotics and autonomous systems*, 42(3-4):143–166, 2003.

[58] David H Fremlin. *Measure theory*, volume 4. Torres Fremlin, 2000.

[59] Luca Galbusera, Marco Pietro Enrico Marciandi, Paolo Bolzern, and Giancarlo Ferrari-Trecate. Control schemes based on the wave equation for consensus in multi-agent systems with double-integrator dynamics. In *Decision and Control, 2007 46th IEEE Conference on*, pages 1498–1503. IEEE, 2007.

[60] Rubén Martín García, Daniel Hernández de la Iglesia, Juan F de Paz, Valderi RQ Leithardt, and Gabriel Villarrubia. Urban search and rescue with anti-pheromone robot swarm architecture. In *2021 Telecoms Conference (ConfTELE)*, pages 1–6. IEEE, 2021.

[61] Veysel Gazi. Swarm aggregations using artificial potentials and sliding-mode control. *IEEE Transactions on Robotics*, 21(6):1208–1214, 2005.

[62] IM Gel'fand and GE Shilov. Generalized functions (academic, new york, 1964), vol. 1. *Google Scholar*, page 117, 1963.

[63] Jerome A Goldstein. *Semigroups of linear operators and applications*. Oxford University Press, Clarendon Press, 1985.

[64] Daniel Grünbaum and Akira Okubo. Modelling social animal aggregations. In *Frontiers in mathematical biology*, pages 296–325. Springer, 1994.

[65] Fred Y Hadaegh, Andrew E Johnson, David S Bayard, Behçet Açıkmeşe, Soon-Jo Chung, and Raman K Mehra. New guidance, navigation, and control technologies for formation flying spacecraft and planetary landing. In *Advances in Control System Technology for Aerospace Applications*, pages 49–80. Springer, 2016.

[66] Jacques Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton university bulletin*, pages 49–52, 1902.

[67] Heiko Hamann and Heinz Wörn. A framework of space–time continuous models for algorithm design in swarm robotics. *Swarm Intelligence*, 2(2-4):209–239, 2008.

[68] Bruce E Hansen. Lecture notes on nonparametrics. *Lecture notes*, 2009.

[69] Christopher Heil. Banach and hilbert space review, 2006.

[70] Charlotte Hemelrijk. *Self-organisation and evolution of biological and social systems*. Cambridge University Press, 2005.

[71] Daniel Henry. *Geometric theory of semilinear parabolic equations*, volume 840. Springer, 2006.

[72] Suranga Hettiarachchi and William M Spears. Distributed adaptive swarm for obstacle avoidance. *International Journal of Intelligent Computing and Cybernetics*, 2(4):644–671, 2009.

[73] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[74] Gerard Houtman. *The FitzHugh-Nagumo Equations on an Unbounded Domain*. PhD thesis, Master thesis, defended on August 25, 2006.

[75] M Ani Hsieh, Vijay Kumar, and Luiz Chaimowicz. Decentralized controllers for shape generation with robotic swarms. *Robotica*, 26(5):691–701, 2008.

[76] Takeshi Ishida. An algorithm applying the self-organizing capabilities of a reaction–diffusion model to control active swarm robots using only adjacent module information. *Journal of Intelligent & Robotic Systems*, 111(2):70, 2025.

[77] Brian Jefferies. *Evolution processes and the Feynman-Kac formula*, volume 353. Springer Science & Business Media, 2013.

[78] Meng Ji and Magnus Egerstedt. Distributed coordination control of multiagent systems while preserving connectedness. *IEEE Transactions on Robotics*, 23(4):693–703, 2007.

[79] R. E. Kalman and J. E. Bertram. Control system analysis and design via the second method of Lyapunov, ii: Discrete-time systems. *Jrnl. of Basic Engineering*, 82:394–400, 1960.

[80] Solmaz S Kia, Bryan Van Scoy, Jorge Cortes, Randy A Freeman, Kevin M Lynch, and Sonia Martinez. Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Systems Magazine*, 39(3):40–72, 2019.

[81] Andreas Kolling, Phillip Walker, Nilanjan Chakraborty, Katia Sycara, and Michael Lewis. Human interaction with robot swarms: A survey. *IEEE Transactions on Human-Machine Systems*, 46(1):9–26, 2016.

[82] Vishaal Krishnan and Sonia Martínez. Self-organization in multi-agent swarms via distributed computation of diffeomorphisms. In *Mathematical Theory of Networks and Systems*, 2016.

[83] Vishaal Krishnan and Sonia Martínez. Distributed control for spatial self-organization of multi-agent swarms. *arXiv preprint arXiv:1705.03109*, 2017.

[84] Yasuhiro Kuriki and Toru Namerikawa. Formation control with collision avoidance for a multi-uav system using decentralized mpc and consensus-based control. *SICE Journal of Control, Measurement, and System Integration*, 8(4):285–294, 2015.

[85] Giovanni Leoni. *A first course in Sobolev spaces*. American Mathematical Soc., 2017.

[86] D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov chains and mixing times*. American Mathematical Soc.

[87] Herbert Levine, Wouter-Jan Rappel, and Inon Cohen. Self-organization in systems of self-propelled particles. *Physical Review E*, 63(1):017101, 2000.

[88] Peter Li and Shing-Tung Yau. On the schrödinger equation and the eigenvalue problem. *Communications in Mathematical Physics*, 88(3):309–318, 1983.

[89] L Lorenzi, A Lunardi, G Metafune, and D Pallara. Analytic semigroups and reaction-diffusion equations. In *Internet Seminar*, volume 2005, 2004.

[90] Arnold Ludwing. Stochastic differential equations: Theory and applications, 1974.

[91] Gian Carlo Maffettone, Lorenzo Liguori, Eduardo Palermo, Mario Di Bernardo, and Maurizio Porfiri. Mixed reality environment and high-dimensional continuification control for swarm robotics. *IEEE Transactions on Control Systems Technology*, 32(6):2484–2491, 2024.

[92] RH Martin. Differential equations on closed subsets of a banach space. *Transactions of the American Mathematical Society*, 179:399–414, 1973.

[93] Mehran Mesbahi and Magnus Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[94] Joe J Monaghan. Smoothed particle hydrodynamics. *Reports on progress in physics*, 68(8):1703, 2005.

[95] Mitio Nagumo. Über die lage der integralkurven gewöhnlicher differentialgleichungen. *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series*, 24:551–559, 1942.

[96] Dharna Nar and Radhika Kotecha. Optimal waypoint assignment for designing drone light show formations. *Results in Control and Optimization*, 9:100174, 2022.

[97] Aleš Nekvinda and Luděk Zajíček. A simple proof of the rademacher theorem. *Časopis pro pěstování matematiky*, 113(4):337–341, 1988.

[98] Y. Nesterov and A. Nemirovsky. *Interior-point Polynomial Methods in Convex Programming*. SIAM, 1994.

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

[100] Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[101] Hans C Öttinger. *Stochastic processes in polymeric fluids: tools and examples for developing simulation algorithms*. Springer Science & Business Media, 2012.

[102] Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[103] Ganapati P Patil. *Weighted distributions*. Wiley Online Library, 2002.

[104] Nicolae H Pavel and Dumitru Motreanu. *Tangency, flow invariance for differential equations, and optimization problems*, volume 219. CRC Press, 1999.

[105] Nicolae H Pavel and Cornellu Ursescu. Flow-invariant sets for autonomous second order differential equations and applications in mechanics. *Nonlinear Analysis: Theory, Methods & Applications*, 6(1):35–74, 1982.

[106] Marco Pavone, Julie C Castillo-Rogez, Issa AD Nesnas, Jeffrey A Hoffman, and Nathan J Strange. Spacecraft/rover hybrids for the exploration of small solar system bodies. In *Aerospace Conference, 2013 IEEE*, pages 1–11. IEEE, 2013.

[107] Amnon Pazy. *Semigroups of linear operators and applications to partial differential equations*, volume 44. Springer Science & Business Media, 2012.

[108] Luciano CA Pimenta, Nathan Michael, Renato C Mesquita, Guilherme AS Pereira, and Vijay Kumar. Control of swarms based on hydrodynamic models. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 1948–1953. IEEE, 2008.

[109] Luciano CA Pimenta, Guilherme AS Pereira, Nathan Michael, Renato C Mesquita, Mateus M Bosque, Luiz Chaimowicz, and Vijay Kumar. Swarm coordination based on smoothed particle hydrodynamics technique. *IEEE Transactions on Robotics*, 29(2):383–399, 2013.

[110] Jie Qi, Rafael Vazquez, and Miroslav Krstic. Multi-agent deployment in 3-d via pde control. *IEEE Transactions on Automatic Control*, 60(4):891–906, 2015.

[111] Wei Ren, Randal W Beard, and Ella M Atkins. Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 27(2):71–82, 2007.

[112] 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.

[113] Murray Rosenblatt et al. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956.

[114] Walter Rudin. Functional analysis. international series in pure and applied mathematics, 1991.

[115] Erol Şahin. Swarm robotics: From sources of inspiration to domains of application. In *International workshop on swarm robotics*, pages 10–20. Springer, 2004.

[116] Hanspeter Schaub, Gordon G Parker, and Lyon B King. Challenges and prospects of coulomb spacecraft formation control. *Journal of Astronautical Sciences*, 52(1):169–193, 2004.

[117] Jeff Shamma. *Cooperative control of distributed multi-agent systems*. John Wiley & Sons, 2008.

[118] Simon J Sheather et al. Density estimation. *Statistical Science*, 19(4):588–597, 2004.

[119] Carlo Sinigaglia, Andrea Manzoni, and Francesco Braghin. Density control of large-scale particles swarm through pde-constrained optimization. *IEEE Transactions on Robotics*, 38(6):3530–3549, 2022.

[120] Carlo Sinigaglia, Andrea Manzoni, Francesco Braghin, and Spring Berman. Robust optimal density control of robotic swarms. *Automatica*, 176:112218, 2025.

[121] Housheng Su, Xiaofan Wang, and Zongli Lin. Flocking of multi-agents with a virtual leader. *IEEE transactions on automatic control*, 54(2):293–307, 2009.

[122] Adam M Tahir and Anshu Narang-Siddarth. Constructive nonlinear approach to coulomb formation control. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 0868, 2018.

[123] Ying Tan and Zhong-yang Zheng. Research advance in swarm robotics. *Defence Technology*, 9(1):18–39, 2013.

[124] Samet Uzun and Nazim Kemal Ure. A probabilistic guidance approach to swarm-to-swarm engagement problem. *arXiv preprint arXiv:2012.01928*, 2020.

[125] Samet Uzun, Nazım Kemal Üre, and Behçet Açıkmeşe. Decentralized state-dependent markov chain synthesis with an application to swarm guidance. *IEEE Transactions on Automatic Control*, 69(9):5759–5774, 2024.

[126] David A Vallado. *Fundamentals of astrodynamics and applications*, volume 12. Springer Science & Business Media, 2001.

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

[128] Xiaoshen Wang. A simple proof of descartes's rule of signs. *The American Mathematical Monthly*, 111(6):525, 2004.

[129] Dominik Wied and Rafael Weißbach. Consistency of the kernel density estimator: a survey. *Statistical Papers*, 53(1):1–21, 2012.

[130] Chenghao Yu, Dengyu Zhang, and Qingrui Zhang. Grf-based predictive flocking control with dynamic pattern formation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 16788–16794. IEEE, 2024.

[131] Wenwu Yu, Guanrong Chen, and Ming Cao. Distributed leader–follower flocking control for multi-agent dynamical systems with time-varying velocities. *Systems & Control Letters*, 59(9):543–552, 2010.

[132] Wenwu Yu, Guanrong Chen, and Ming Cao. Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems. *Automatica*, 46(6):1089–1095, 2010.

[133] Sheng Zhao, Subramanian Ramakrishnan, and Manish Kumar. Density-based control of multiple robots. In *Proceedings of the 2011 American Control Conference*, pages 481–486. IEEE, 2011.

[134] Tongjia Zheng, Qing Han, and Hai Lin. Transporting robotic swarms via mean-field feedback control. *arXiv preprint arXiv:2006.11462*, 2020.

[135] Tongjia Zheng, Zhiyu Liu, and Hai Lin. Complex pattern generation for swarm robotic systems using spatial-temporal logic and density feedback control. In *2020 American Control Conference (ACC)*, pages 5301–5306. IEEE, 2020.

[136] Minghui Zhu and Sonia Martínez. *Distributed Optimization-Based Control of Multi-Agent Networks in Complex Environments*. Springer, 2015.

Appendix A

PROBABILISTIC GUIDANCE ALGORITHM (PGA)

Assuming that each agent can determine its current state and has an algorithm to sample a random number from a uniform distribution, the PGA [10] can be implemented by using M_n as follows:

Algorithm 1: Probabilistic Guidance Algorithm (PGA)

Given $\mathbf{r}_1(0), \ldots, \mathbf{r}_N(0)$, set $k = 0$;
Each agent performs the following;
while *true* **do**
  Agent determines its current subregion, e.g., $\mathbf{r}_n(k) \in R_i$;
  Agent generates a random number z that is uniformly distributed in $[0, 1]$;
  if $z \leq M_n[1, i]$ **then**
    $j = 1$;
  else
    find $q \in \{2, \ldots, m\}$ such that, $\sum_{l=1}^{q-1} M_n[l, i] \leq z \leq \sum_{l=1}^{q} M_n[l, i]$;
    $j = q$;
  end
  Agent transitions from subregion R_i to R_j, i.e., $\mathbf{r}_n(k+1) \in R_j$;
  $k \leftarrow k + 1$
end

With the definition of PGA we can state the following lemma that provides two interpretations for the swarm mass distribution $\theta(k)$: (i) $\theta(k)$ is the vector of expected ratios of the number of agents in each subregion, (ii) the ensemble of agent states, $\{\mathbf{r}_n(k)\}_{n=1}^{N}$, has a distribution that approaches $\theta(k)$ with probability one as $N \to \infty$.

Lemma 11 *Consider N agents where $\theta_1(0) = \ldots = \theta_N(0) = \theta_0 \in \mathbb{P}^m$ with $\theta_n,\ n = \{1, \ldots, N\}$ defined as in (2.6). Further, suppose that each agent uses the PGA algorithm with the same*

Markov matrix $M(k)$, that is, $M_1(k) = \ldots = M_N(k) = M(k)$. Then, $\theta_1(k) = \ldots = \theta_N(k) = \theta(k)$ for $k = 0, 1, \ldots,$ where

$$\theta(k+1) = M(k)\theta(k), \quad \textit{with } \theta(0) = \theta_0, \text{ and} \tag{A.1}$$

$$\theta[i](k) = \text{prob}(\mathbf{r}(k) \in R_i) = \mathbb{E}\left[\frac{\mathbf{n}[i](k)}{N}\right], \quad i = 1, \ldots, m, \tag{A.2}$$

$\text{prob}(\mathbf{r}(k) \in R_i)$ *is the probability of finding an agent in the i^{th} subregion, and $\mathbf{n}(k)$ is the vector of the number of agent states in each subregion at time index k. Furthermore,*

$$\text{prob}\left(\lim_{N\to\infty} \frac{\mathbf{n}(k)}{N} = \theta(k)\right) = 1, \quad k = 0, 1, \ldots. \tag{A.3}$$

Proof: It is straight forward to show that (2.11) implies (A.1) by using the the Total Probability theorem [34] and noting that $\text{prob}(\mathbf{r}(k+1) \in R_i) = \sum_{j=1}^{m} \text{prob}(\mathbf{r}(k+1) \in R_i | \text{prob}(\mathbf{r}(k) \in R_j))\text{prob}(\mathbf{r}(k) \in R_j)$. Since the PGA algorithm uses the same Markov matrix for each agent at each time, $M_1(k) = \ldots = M_N(k) = M(k)$ with $\theta_1(0) = \ldots = \theta_N(0) = \theta(0)$, the probability distribution of all agents evolves according to (A.1). Clearly $\text{prob}(\mathbf{r}(k) \in R_i)$ is the probability of finding any of the agents in i^{th} subregion. Since $\mathbb{E}\big[\mathbf{n}[i](k)\big] = \sum_{n=1}^{N} \theta_n[i](k) = N\theta[i](k)$, we have $\theta(k) = \mathbb{E}\big[\mathbf{n}(k)/N\big]$.

Next, we prove (A.3) by using standard arguments as in Theorem 5.4.2 of [34]. Consider $\theta[i](k)$ which is the probability of finding any agent in R_i at time index k. Consider a new Random Variable (RV), $Z_n[i](k)$, such that $Z_n[i](k) = 1$ if the agent n is in R_i at time index k, and zero otherwise. Clearly, $\mathbb{E}\big[Z_n[i](k)\big] = \theta_n[i](k) = \theta[i](k)$ and $Z_n[i](k)$, $n = 1, ..., N$, form Independently Identically Distributed (iid) RVs. Then it follows that $\mathbf{n}[i](k) = Z_1[i](k) + \ldots + Z_N[i](k)$. Since $Z_n[i](k)$, $n = 1, ..., N$ are iid RVs and $\mathrm{E}\big[|Z_n[i](k)|\big] < \infty$, we can use the strong law of large numbers theorem, Theorem 5.4.2 [34], to conclude, $\text{prob}\left(\lim_{N\to\infty} \dfrac{\mathbf{n}[i](k)}{N} = \theta[i](k)\right) = 1$, which implies (A.3). ■

It follows from the independent and identically distributed agent realizations that $\theta(k) =$

$\mathbb{E}\big[\mathbf{n}(k)\big]/N$, hence, the actual distribution of agent states $\mathbf{n}(k)/N$ will generally be different from $\theta(k)$. However, $\mathbf{n}(k)/N$ can be made arbitrarily close to $\theta(k)$ with large number of agents [10].

Appendix B
HEAT EQUATION BASED CONTROL

B.1 Exponential Convergence

Before giving the proof of Theorem 12 which is provided in [135, 134] with different versions, the following lemmas will be used in the main result.

Lemma 12 (Divergence Theorem) *For a smooth vector field $\boldsymbol{F}$ over a bounded open set $\mathcal{R} \subseteq \mathbb{R}^d$ with boundary $\partial\mathcal{R}$, the the integral of the divergence $\nabla \cdot \boldsymbol{F}$ over $\mathcal{R}$ is equal to the surface integral of $\boldsymbol{F}$ over $\partial\mathcal{R}$:*

$$\int_{\mathcal{R}} \nabla \cdot \boldsymbol{F} \, \mathrm{d}\mathbf{x} \; = \; \int_{\partial\mathcal{R}} \boldsymbol{F} \cdot \boldsymbol{n} \, \mathrm{dS} \tag{B.1}$$

where $\boldsymbol{n}$ is the outward normal vector to the boundary $\partial\mathcal{R}$, and dS is the measure on the boundary. For a scalar field u and a vector field $\boldsymbol{F}$,

$$\int_{\mathcal{R}} u\big(\nabla \cdot \boldsymbol{F}\big) \, \mathrm{d}\mathbf{x} \; = \; \int_{\partial\mathcal{R}} u\big(\boldsymbol{F} \cdot \boldsymbol{n}\big) \, \mathrm{dS} \; - \; \int_{\mathcal{R}} \nabla u \cdot \boldsymbol{F} \, \mathrm{d}\mathbf{x}. \tag{B.2}$$

Lemma 13 (Poincaré Inequality) *[85] For $p \in [1, \infty)$ and $\mathcal{R}$ a bounded connected open subset of $\mathbb{R}^d$ with a Lipschitz boundary, there exists a constant C depending only on $\mathcal{R}$ and p such that for every function $f \in W^{1,p}(\mathcal{R})$,*

$$\|f - f_{\mathcal{R}}\|_{L^p(\mathcal{R})} \; \leq \; C\|\nabla f\|_{L^p(\mathcal{R})} \tag{B.3}$$

where $f_{\mathcal{R}} \; = \; \frac{1}{|\mathcal{R}|} \int_{\mathcal{R}} f \, \mathrm{d}\mathbf{x}$, and $|\mathcal{R}|$ is the Lebesgue measure of $\mathcal{R}$.

Now assuming that *Wiener process* is a full state vector, i.e., $\mathbf{W}(t) \in \mathbb{R}^d$ and diffusion function is a scalar field i.e., $\sigma(t, \mathbf{x}) \in \mathbb{R}$ the exponential convergence and stability result can be presented as follows.

Theorem 12 *[135, 134] Consider the following velocity field for the swarm $\mathcal{S}(t)$,*

$$\dot{\mathbf{x}}(t) = -\frac{\alpha(t, \mathbf{x})\nabla\Phi(t, \mathbf{x}) - \nabla\big(\sigma^2(t, \mathbf{x})\, f_{\mathcal{R}}(t, \mathbf{x})/2\big)}{f_{\mathcal{R}}(t, \mathbf{x})}. \tag{B.4}$$

where $\alpha(t, \mathbf{x}) > 0$ is the diffusion parameter that satisfies,

$$\sup_{(t,\mathbf{x})} \alpha(t, \mathbf{x}) < \infty \qquad \text{and} \qquad \inf_{(t,\mathbf{x})} \alpha(t, \mathbf{x}) > 0.$$

If the solution satisfies $f_{\mathcal{R}}(t, \mathbf{x}) > 0$ for all $t > 0$, then $\|\Phi(t, \mathbf{x})\|_{L^2(\mathcal{R})} \to 0$ exponentially.

Proof: First observe that for scalar field diffusion function $\sigma(t, \mathbf{x}) \in \mathbb{R}$ the *Fokker-Planck equation* (3.4) simplifies to:

$$\dot{f}_{\mathcal{R}}(t, \mathbf{x}) = -\nabla \cdot \big(\mathbf{v}(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big) + \frac{1}{2}\nabla \cdot \nabla\big(\sigma^2(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big). \tag{B.5}$$

now substituting the control law defined by (B.4) into (B.5):

$$\begin{aligned}
\dot{f}_{\mathcal{R}}(t, \mathbf{x}) &= \nabla \cdot \big(\mathbf{v}(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big) + \frac{1}{2}\nabla \cdot \nabla\big(\sigma^2(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big) && \text{(B.6)}\\
&= \nabla \cdot \Big(\alpha(t, \mathbf{x})\nabla\Phi(t, \mathbf{x}) - \frac{1}{2}\nabla\big(\sigma^2(t, \mathbf{x})\, f_{\mathcal{R}}(t, \mathbf{x})\big)\Big) + \frac{1}{2}\nabla \cdot \nabla\big(\sigma^2(t, \mathbf{x}) f_{\mathcal{R}}(t, \mathbf{x})\big) && \text{(B.7)}\\
&= \nabla \cdot \big(\alpha(t, \mathbf{x})\nabla\Phi(t, \mathbf{x})\big) && \text{(B.8)}
\end{aligned}$$

again considering time-invariant desired density, the equation (B.5) becomes a *non-linear heat equation*:

$$\dot{\Phi}(t, \mathbf{x}) = \nabla \cdot \big(\alpha(t, \mathbf{x})\nabla\Phi(t, \mathbf{x})\big) \tag{B.9}$$

with a *Neumann boundary condition* of $\nabla\Phi(t, \mathbf{x}) = 0, \ \forall \mathbf{x} \in \partial\mathcal{R}$.

Now consider a Lyapunov function:

$$V(t) = \frac{1}{2}\|\Phi(t,\mathbf{x})\|^2_{L^2(\mathcal{R})} = \frac{1}{2}\int_{\mathcal{R}} \Phi^2(t,\mathbf{x})\ \mathrm{d}\mathbf{x}, \tag{B.10}$$

and define $\alpha_{min}(t) = \inf_{\mathbf{x}\in\mathcal{R}} \alpha(t,\mathbf{x}) > 0,\ \forall t \in [0,\infty)$, then we have,

$$\begin{aligned}
\dot{V}(t) &= \int_{\mathcal{R}} \Phi(t,\mathbf{x})\dot{\Phi}(t,\mathbf{x})\ \mathrm{d}\mathbf{x} = \int_{\mathcal{R}} \Phi(t,\mathbf{x})\nabla\cdot\big(\alpha(t,\mathbf{x})\nabla\Phi(t,\mathbf{x})\big)\ \mathrm{d}\mathbf{x} \\
&= \int_{\partial\mathcal{R}} \Phi(t,\mathbf{x})\big(\alpha(t,\mathbf{x})\nabla\Phi(t,\mathbf{x})\cdot\mathbf{n}\big)\,\mathrm{dS} - \int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})\cdot\alpha(t,\mathbf{x})\nabla\Phi(t,\mathbf{x})\,\mathrm{d}\mathbf{x} \quad \text{(divergence)} \\
&= -\int_{\mathcal{R}} \nabla\Phi(t,\mathbf{x})\cdot\alpha(t,\mathbf{x})\nabla\Phi(t,\mathbf{x})\,\mathrm{d}\mathbf{x} \qquad \text{(Neumann BC)} \\
&\le -\alpha_{min}(t)\int_{\mathcal{R}} \|\nabla\Phi(t,\mathbf{x})\|_2^2\,\mathrm{d}\mathbf{x} \\
&\le -\frac{\alpha_{min}(t)}{C^2}\int_{\mathcal{R}} |\Phi(t,\mathbf{x})|^2\,\mathrm{d}\mathbf{x} \qquad \text{(Poincaré Inequality)}
\end{aligned}$$

Note that last inequality uses the fact that $\Phi_{\mathcal{R}}(t) = \int_{\mathcal{R}} \Phi(t,\mathbf{x})\mathrm{d}\mathbf{x} = 0$, and implies exponential convergence. ■

B.2 *Finite-Agent Analysis*

Under the dynamics (5.19), the density error cost function $\Phi(\mathcal{X}, h)$ is dissipated in time, i.e.,

$$\dot{\Phi}(\mathcal{X},h) \ = \ \sum_{i=1}^{N} \nabla^{\intercal}_{\mathbf{x}_i}\Phi(\mathcal{X},h)\,\dot{\mathbf{x}}_i \ = \ -\alpha\sum_{i=1}^{N}\frac{\|\nabla_{\mathbf{x}_i}\Phi(\mathcal{X},h)\|^2}{\rho(\mathbf{x}_i)} \ \le \ 0, \tag{B.11}$$

which implies that the value $\Phi(\mathcal{X}, h)$ is non-increasing, hence all the sublevel sets $\{\hat{\mathcal{X}} \in \mathbb{R}^{Nd} \mid \Phi(\hat{\mathcal{X}}, h) \le C\}$ are invariant. Consequently, trajectories initialized in a bounded sublevel set remain bounded for all future time.

Lemma 14 *If $f(t)$ is continuous on $a \le t < \infty$ and $\int_a^\infty f(t)dt$ converges, then* $\lim_{t\to\infty} f(t) =$ 0 *if, and only if, $f(t)$ is uniformly continuous on $a \le t < \infty$.*

One can show the stability & convergence of the system (5.19) as follows.

Theorem 13 (Stability & Convergence) *Given any initial condition* $\mathbf{x}_1(0), \ldots, \mathbf{x}_N(0) \in \mathbb{R}^d$*, the solution of the system (5.19) satisfies:*

$$\lim_{t\to\infty} \nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h) = 0 \quad \forall i, \tag{B.12}$$

for kernel functions where $K(\mathbf{x})$ *and* $\nabla K(\mathbf{x})$ *are both differentiable and Lipschitz continuous.*

Proof: Note that $0 < \epsilon \leq \rho(\mathbf{x}_i) \leq \bar{\rho} < \infty$ $\forall i \in \{1, \ldots, N\}$ by kernel density estimation (5.12) and $0 < \alpha$. For differentiable and Lipschitz continuous $K(\mathbf{x})$ and $\nabla K(\mathbf{x})$, it follows that the function:

$$F(\mathcal{X}, h) = \alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h)\|^2}{\rho\big(\mathbf{x}_i\big)}$$

is differentiable and has bounded gradient i.e. $\|\nabla_{\mathcal{X}} F(\mathcal{X}, h)\| \leq C$, hence it is also Lipschitz continuous in $\mathcal{X}$ argument. Since Lipschitz continuity implies uniform continuity, we will investigate the convergence of the following integral and use Lemma 14:

$$\eta \;=\; \int_0^\infty F(\mathcal{X}, h)\; dt \;=\; \int_0^\infty \alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h)\|^2}{\rho\big(\mathbf{x}_i\big)}\; dt.$$

using (B.11) one has,

$$\begin{aligned}
\int_0^t \alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i}\Phi(\mathcal{X}, h)\|^2}{\rho\big(\mathbf{x}_i\big)}\; d\tau \;=\; \int_0^t -\dot{\Phi}(\mathcal{X}(\tau), h)\; d\tau \;&=\; \Phi(\mathcal{X}(0), h) - \Phi(\mathcal{X}(t), h) \\
&\leq\; \sup_t \Big(\Phi_0 - \Phi(\mathcal{X}(t), h)\Big) \\
&=\; \Phi_0 - \inf_t \Big(\Phi(\mathcal{X}(t), h)\Big) \\
&=\; \Phi_0 - \inf_{\hat{\mathcal{X}} \in \Omega} \big(\Phi(\hat{\mathcal{X}}, h)\big) \\
&\leq\; \Phi_0 < \infty,
\end{aligned}$$

where $\Omega = \left\{ \mathcal{X} \in \mathbb{R}^{Nd} \mid \Phi(\mathcal{X}, h) \leq \Phi_0 \right\}$. Therefore it follows from Lemma 14 that,

$$\eta < \infty \quad \Rightarrow \quad \lim_{t\to\infty} \alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h)\|^2}{\rho(\mathbf{x}_i)} = 0 \quad \Rightarrow \quad \lim_{t\to\infty} \nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h) = 0, \quad \forall i \in \{1, \ldots, N\}.$$

■

Theorem 13 establishes *Lyapunov stability* of the sublevel sets of $\Phi(\mathcal{X}, h)$ together with convergence to the critical set characterized by $\nabla_{\mathcal{X}} \Phi(\mathcal{X}, h) = 0$. However, asymptotic convergence toward minimizing equilibria requires additional assumptions on the structure of the critical points $\mathcal{X}^\star = \{\mathcal{X} \in \mathbb{R}^{Nd} \mid \nabla_{\mathcal{X}} \Phi(\mathcal{X}, h) = 0\}$. To be more precise, critical points need to be *isolated local minima. Strict local minimum* is a sufficient condition when $\Phi(\mathcal{X}, h)$ is analytic function, i.e., it is C^∞ and at all points in its domain, function converges to its Taylor Series approximation. If $\Phi(\mathcal{X}, h)$ is an analytic function then one can invoke *Lojasiewicz's inequality* which is given as follows.

Lemma 15 (Lojasiewicz's inequality) *Let f be a real analytic function on a neighbourhood of $z \in \mathbb{R}^n$. Then there are constants $c > 0$ and $a \in [0, 1)$ such that*

$$\|\nabla f(x)\| \geq c|f(x) - f(z)|^a$$

in some neighborhood of z.

Now consider $\bar{\mathcal{X}} \in \mathcal{X}^\star$ and define $V(\mathcal{X}) = \Phi(\mathcal{X}, h) - \Phi(\bar{\mathcal{X}}, h) \geq 0$ which is defined in the neighborhood of $\bar{\mathcal{X}}$ and is an analytic function. Since $0 < \epsilon \leq \rho(\mathbf{x}_i) \leq \bar{\rho} < \infty \;\; \forall i$, one has the following,

$$\begin{aligned}
\dot{V}(\mathcal{X}) &= -\alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h)\|^2}{\rho\big(\mathbf{x}_i\big)} \;\leq\; -\alpha \sum_{i=1}^{N} \frac{\|\nabla_{\mathbf{x}_i} \Phi(\mathcal{X}, h)\|^2}{\bar{\rho}} \\
&= -\frac{\alpha}{\bar{\rho}} \|\nabla_{\mathcal{X}} \Phi(\mathcal{X}, h)\|^2 \\
&\leq -\frac{\alpha c^2}{\bar{\rho}} \left|\Phi(\mathcal{X}, h) - \Phi(\bar{\mathcal{X}}, h)\right|^{2a} \\
&= -C\, V(\mathcal{X})^{2a}.
\end{aligned}$$

The last inequality follows from Lemma 15. Therefore the closed-loop system satisfies $\dot{V}(\mathcal{X}) \leq -CV(\mathcal{X})^{2a}$, which implies local asymptotic convergence toward an equilibrium point $\bar{\mathcal{X}} \in \mathcal{X}^{\star}$ with convergence rate determined by the Lojasiewicz's exponent a. In particular, when $a = 1/2$, the equilibrium is locally exponentially stable.

Appendix C

DENSITY AS KERNEL MOLLIFICATION OF EMPIRICAL MEASURES

Observation 3 *Given agent locations $\mathbf{x}_i(t) \in \mathbb{R}^d$ $\forall i \in \{1, \ldots, N\}$, define an empirical measure (generalized function) for the agent locations*

$$\mathcal{I}_D(t, \mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \delta\big(\mathbf{x} - \mathbf{x}_i(t)\big)$$

then the density estimation (or physical density approximation) given with (3.10) can be written in the convolution form as follows:

$$\begin{aligned}
\rho(t, \mathbf{x}) = \mathcal{I}_D(t, \mathbf{x}) * K_h(\mathbf{x}) &= \int_{\mathbb{R}^d} \mathcal{I}_D(t, \tau) K_h(\mathbf{x} - \tau) \ \mathrm{d}\tau \\
&= \frac{1}{N} \int_{\mathbb{R}^d} \sum_{i=1}^{N} \delta\big(\tau - \mathbf{x}_i(t)\big) K_h(\mathbf{x} - \tau) \ \mathrm{d}\tau \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_{\mathbb{R}^d} \delta\big(\tau - \mathbf{x}_i(t)\big) K_h(\mathbf{x} - \tau) \ \mathrm{d}\tau \\
&= \frac{1}{N} \sum_{i=1}^{N} K_h\big(\mathbf{x} - \mathbf{x}_i(t)\big),
\end{aligned} \tag{C.1}$$

where $K_h(\mathbf{x})$ is defined with the assumption of equal smoothing parameters $h > 0$ for each dimension as,

$$K_h(\mathbf{x}) = h^{-d} K\Big(\frac{\mathbf{x}}{h}\Big)$$

Therefore, the density estimation is obtained as the convolution of an interaction kernel with the empirical measure supported at the agent locations. Here, the empirical measure is interpreted in the sense of distributions. From this point on, the following notation is used,

$$\rho(\mathbf{x},\mathcal{X}) = \rho(\mathbf{x},\mathbf{x}_1,\ldots,\mathbf{x}_N) := \rho(t,\mathbf{x}),$$
$$\mathcal{I}_D(\mathbf{x},\mathcal{X}) = \mathcal{I}_D(\mathbf{x},\mathbf{x}_1,\ldots,\mathbf{x}_N) := \mathcal{I}_D(t,\mathbf{x}).$$

Empirical measure $\mathcal{I}_D(\mathbf{x},\mathcal{X})$ is not absolutely continuous with respect to a Riemannian measure whereas the convolution form approximation $\rho(\mathbf{x},\mathcal{X})$ does satisfy this property and has the notion of density given $\int_{\mathbb{R}^d} K_h(\mathbf{x})\mathrm{d}\mathbf{x} = 1$. Dirac Delta $\delta(\mathbf{x})$ is a generalized function, that is, a continuous linear functional defined on the space of all real functions with continuous derivative of all orders and compact support [62]. The convolution with Dirac measure is well defined because δ is concentrated at a point (see p. 104 in [62]).

Lemma 16 ***(Theorem 6.37 (d)-(e) in*** *[114]) Suppose u and v are generalized functions with supports S_u and S_v,*

1. *If δ is the Dirac measure, then $D^\alpha u = D^\alpha \delta * u$.*

2. *If at least one of the sets S_u and S_v is compact, then $D^\alpha(u*v) = (D^\alpha u)*v = u*(D^\alpha v)$*

for every multi-index α which denotes an ordered n-tuple $\alpha = (\alpha_1,\ldots,\alpha_n)$ of non-negative integers α_i with each multi-index is associated the differential operator

$$D^\alpha = \left(\frac{\partial}{\partial \xi_1}\right)^{\alpha_1} \cdots \left(\frac{\partial}{\partial \xi_d}\right)^{\alpha_n}$$

whose order is $|\alpha| = \sum \alpha_i$. If $|\alpha| = 0$ then, $D^\alpha u = u$.

Using Lemma 16 and assuming that $K_h(\mathbf{x}) \in C^1(\mathcal{R},\mathbb{R})$, we can state the following for the density gradient estimations:

$$\nabla_{\mathbf{x}}\big(\rho(\mathbf{x},\mathcal{X})\big) = \nabla_{\mathbf{x}}\big(\mathcal{I}_D(\mathbf{x},\mathcal{X}) * K_h(\mathbf{x})\big) = \mathcal{I}_D(\mathbf{x},\mathcal{X}) * \nabla K_h(\mathbf{x}).$$

This construction provides a rigorous link between finite agent configurations and smooth density fields used throughout the dissertation.

Appendix D

PRELIMINARIES ON FUNCTIONAL ANALYSIS & SEMIGROUPS

This appendix summarizes standard results from functional analysis, semigroup theory, and stability theory used in the dissertation. Detailed proofs and classical background are included here for completeness.

D.1 Functional Spaces and Operators

Definition 12 (Cauchy sequence) *Let (X, d) be a metric space, i.e., non-empty set X together with a metric d on the X. The sequence $(x_n)_{n\in\mathbb{N}} \subseteq X$ is a Cauchy sequence if for all $\epsilon > 0$ there exists an $N_\epsilon \in \mathbb{N}$ such that for all $n, m \geq N_\epsilon$, $d(x_n, x_m) < \epsilon$.*

Definition 13 (Complete Metric Space) *A metric space (X, d) is complete if every Cauchy sequence in X has a limit in X, i.e., every Cauchy sequence is convergent in X with respect to the metric d.*

Definition 14 (Banach Space) *A Banach space X is a normed vector space that is complete with respect to the metric induced by its norm, i.e. $d(x, y) = \|x - y\|_X$.*

The following are some of the function spaces for $1 \leq p < \infty$ which are *Banach* under the given norms [52, 69]:

$$
\begin{aligned}
L^p(\mathbb{R}^d, X) &= \left\{ f : \mathbb{R}^d \to X : \int_{\mathbb{R}^d} \|f(x)\|_X^p \, dx < \infty \right\}, \quad \|f\|_{L^p(\mathbb{R}^d,X)} = \left(\int_{\mathbb{R}^d} \|f(x)\|_X^p \, dx \right)^{1/p} \\
L^\infty(\mathbb{R}^d, X) &= \left\{ f : \mathbb{R}^d \to X : f \text{ is essentially bounded} \right\}, \quad \|f\|_{L^\infty(\mathbb{R}^d,X)} = \operatorname*{ess\,sup}_{x \in \mathbb{R}^d} \|f(x)\|_X \\
C_b(\mathbb{R}^d, X) &= \left\{ f \in L^\infty(\mathbb{R}^d, X) : f \text{ is bounded, continuous} \right\}, \quad \|f\|_{L^\infty(\mathbb{R}^d,X)} = \sup_{x \in \mathbb{R}^d} \|f(x)\|_X \\
C_0(\mathbb{R}^d, X) &= \left\{ f \in C_b(\mathbb{R}^d, X) : \lim_{\|x\| \to \infty} f(x) = 0 \right\}, \quad \|f\|_{L^\infty(\mathbb{R}^d,X)} = \sup_{x \in \mathbb{R}^d} \|f(x)\|_X
\end{aligned}
$$

A closed subspace of a Banach space X is itself a Banach space under the norm of X. Conversely, if a subspace of X is a Banach space under the norm of X, then it is a closed subspace of X. Therefore, $C_b(\mathbb{R}^d, X)$ and $C_0(\mathbb{R}^d, X)$ are closed subspaces of $L^\infty(\mathbb{R}^d, X)$ under the $\|f\|_{L^\infty(\mathbb{R}^d,X)}$ [69]. Typically, for $X = \mathbb{R}$ second argument is omitted in the definition, e.g., $L^p(\mathbb{R}^d)$. Another Banach space of interest within this context is the *Sobolev* space which is defined for $1 \leq p \leq \infty$, $\alpha \in \mathbb{N}_0^d$, and $k \in \mathbb{N}_0$ as follows:

$$
W^{k,p}(\mathbb{R}^d) = \left\{ f \in L^p(\mathbb{R}^d) : D^\alpha f \in L^p(\mathbb{R}^d) \text{ for all } |\alpha| \leq k \right\}, \quad \|f\|_{W^{k,p}(\mathbb{R}^d)} \coloneqq \sum_{|\alpha| \leq k} \|D^\alpha f\|_{L^p(\mathbb{R}^d)}
$$

If $p = 2$, it is customary to write $H^k(\mathbb{R}^d) \coloneqq W^{k,2}(\mathbb{R}^d)$, which denotes *Hilbert* space with the norm $\| \cdot \|_{W^{k,2}}$ [54]. Note that $H^0(\mathbb{R}^d) = L^2(\mathbb{R}^d)$.

The following are subspaces of $L^\infty(\mathbb{R}^d)$, but they are not closed subspaces, hence they are <u>not</u> Banach spaces under $\| \cdot \|_{L^\infty(\mathbb{R}^d)}$,

$$
\begin{aligned}
C_c(\mathbb{R}^d) &= \left\{ f \in C_b(\mathbb{R}^d) \ : \ f \text{ has compact support} \right\}, \\
C_b^\infty(\mathbb{R}^d) &= \left\{ f \in C_b(\mathbb{R}^d) \ : \ f \text{ is infinitely differentiable, } D^\alpha f \in L^\infty(\mathbb{R}^d) \text{ for all } \alpha \in \mathbb{N}_0^d \right\}, \\
\mathbb{S}(\mathbb{R}^d) &= \left\{ f \in C_b^\infty(\mathbb{R}^d) \ : \ \lim_{\|x\| \to \infty} \|x\|^k D^\alpha f(x) = 0, \text{ for all } k \in \mathbb{N}_0 \text{ and } \alpha \in \mathbb{N}_0^d \right\},
\end{aligned}
$$

The space $\mathbb{S}(\mathbb{R}^d)$ is called the *Schwartz* space of rapidly decreasing functions.

Definition 15 (Bounded Linear Operator) *Let $(X, \|\cdot\|_X)$ and $(Y, \|\cdot\|_Y)$ be normed spaces. The operator $T : X \to Y$ is a bounded linear operator if T is linear, and there exists $M > 0$ such that*

$$\|Tx\|_Y \leq M\|x\|_X \qquad \text{for all } x \in X.$$

The space of bounded linear operators is denoted as $\mathcal{L}(X, Y)$ and it is equipped with the operator norm that is defined by,

$$\|T\|_{\mathcal{L}(X,Y)} := \inf\big\{M : \|Tx\|_Y \leq M\|x\|_X \text{ for all } x \in X\big\}, \qquad T \in \mathcal{L}(X, Y).$$

Note that, $\mathcal{L}(X)$ is the space of bounded linear operators from a normed space $(X, \|\cdot\|_X)$ to itself.

D.2 Semigroups

Following definitions and results can be found in [107, 51, 52]

Definition 16 (Semigroup) *A semigroup on a complete metric space (X, d) is a family of maps $\{S(t) : X \to X,\ t \geq 0\}$ such that,*

(i) $S(0)$ is the identity on X;

(ii) $S(t)\big(S(\tau)x\big) = S(t + \tau)x$ for all $x \in X$ and $t, \tau \geq 0$;

(iii) for each $t \geq 0$, $S(t) : X \to X$ is continuous with respect to d.

Definition 17 (C_0-semigroup) *A semigroup of bounded linear operators is strongly continuous if for all $x \in X$, $t \to S(t)x$ is continuous for $t \geq 0$. Due to (ii), continuity on $t = 0$ is sufficient for strong continuity (Proposition 1.3 [52]), i.e.,*

$$\lim_{t\to 0^+} S(t)x = x \qquad \text{for each } x \in X \text{ with respect to } d.$$

A strongly continuous semigroup of bounded linear operators on X is called C_0-semigroup.

Definition 18 (Infinitesimal Generator) *The infinitesimal generator of a C_0-semigroup is an operator A_S defined by,*

$$A_S x = \lim_{t \to 0^+} \frac{1}{t}\big(S(t)x - x\big) \qquad \text{for } x \in \mathcal{D}(A_S)$$

where $\mathcal{D}(A_S)$ is the domain of the generator consisting of all functions $x \in X$ for which this limit in X exists, i.e.,

$$\mathcal{D}(A_S) = \left\{ x \in X : \lim_{t \to 0^+} \frac{1}{t}\big(S(t)x - x\big) \text{ exists} \right\}.$$

Note that each $S(t) : X \to X$ is linear if A_S is linear.

Proposition 7 *Let $\{S(t) : X \to X,\ t \geq 0\}$ be a C_0-semigroup on a Banach space X with generator A_S.*

(a) $S(t)$ is bounded, i.e., there exist constants $\omega \geq 0$ and $M \geq 1$ such that $\|S(t)\| \leq Me^{\omega t}$ for all $t \geq 0$.

(b) The domain $\mathcal{D}(A_S)$ is dense in X, i.e., $\overline{\mathcal{D}(A_S)} = X$ and A_S is a closed operator.

(c) $\mathcal{D}(A_S)$ is invariant under the semigroup, i.e., for all $x \in \mathcal{D}(A_S)$ and $t \geq 0$, $S(t)x \in \mathcal{D}(A_S)$.

(d) The generator and the semigroup commute, i.e., for all $x \in \mathcal{D}(A_S)$ and $t \geq 0$, $A_S S(t)x = S(t)A_S x$.

(e) For all $x \in \mathcal{D}(A_S)$, $t \to S(t)x$ is continuously differentiable for $t \geq 0$, and

$$\frac{d}{dt}S(t)x = A_S S(t)x = S(t)A_S x, \qquad t \geq 0.$$

Definition 19 (Core for a Generator) *Let $\{S(t) : X \to X,\ t \geq 0\}$ be a C_0-semigroup on a Banach space X with generator $A_S : D(A_S) \subset X \to X$. A subspace D of $\mathcal{D}(A_S)$ is called a core for A_S if D is invariant under the semigroup $\{S(t) : X \to X,\ t \geq 0\}$, and D is dense in $\mathcal{D}(A_S)$ with respect to the graph norm $\|x\|_{A_S} := \|x\|_X + \|A_S x\|_X$.*

Theorem 14 (Uniqueness of C_0-semigroup) *The generator of a C_0-semigroup is a closed and densely defined linear operator that determines the semigroup uniquely.*

Theorem 15 (Bounded Perturbation) *Let $\{S(t) : X \to X,\ t \geq 0\}$ be a C_0-semigroup on a Banach space X with generator A_S. Let B be a bounded linear operator, i.e. $B \in \mathcal{L}(X)$, then the operator,*

$$\tilde{A}_S := A_S + B \qquad \text{with the domain} \qquad \mathcal{D}(\tilde{A}_S) := \mathcal{D}(A_S)$$

generates a C_0-semigroup $\{\tilde{S}(t) : X \to X,\ t \geq 0\}$ and there exist constants $\omega \geq 0$ and $M \geq 1$ such that

$$\|\tilde{S}(t)\| \leq M e^{(\omega + M\|B\|)t} \qquad \text{for all } t \geq 0.$$

D.2.1 Abstract Cauchy Problem

Consider the abstract Cauchy problem (or initial value problem)

$$\begin{aligned} \frac{d\Theta(t)}{dt} &= A\Theta(t), \qquad t > 0 \\ \Theta(0) &= \Theta_0, \end{aligned} \tag{D.1}$$

where $\Theta_0 \in \mathcal{D}(A)$, and A is an operator from its domain $\mathcal{D}(A)$ in X to X. Assume X is a Banach space, then the equation $\dot{\Theta} = A\Theta$ is interpreted to mean that $\Theta(t)$ belongs to $\mathcal{D}(A)$ and that,

$$\lim_{h \to 0} \left\| h^{-1}[\Theta(t+h) - \Theta(t)] - A\Theta(t) \right\| = 0,$$

where $\|\cdot\|$ denotes the norm in X [63]. The following result gives the conditions for *well-posedness* and *uniqueness* of the solution for (D.1).

Theorem 16 (Well-posedness & Uniqueness Theorem I in [63]) *The abstract Cauchy problem given by (D.1) (with A is linear) is well-posed iff A is the generator of a (C_0) semigroup $\{S(t) : X \to X,\ t \geq 0\}$. In this case, there exists a unique solution $\Theta : \mathbb{R}_+ \to \mathcal{D}(A)$ of (D.1) such that $\Theta \in C^1(\mathbb{R}_+, X)$ and it is explicitly given by $\Theta(t) = S(t)\,\Theta_0$, for any $\Theta_0 \in \mathcal{D}(A)$.*

D.2.2 Stability of Semigroups

Definitions and results given in section can be found in [71].

Definition 20 (Dynamical System, Definition 4.1.1) *A dynamical system (nonlinear semigroup) on a complete metric space (C, d) is a family of maps $\{S(t) : C \to C,\ t \geq 0\}$ such that,*

(i) for each $t \geq 0$, $S(t)$ is continuous from C to C;

(ii) for each $x \in C$, $t \to S(t)x$ is continuous;

(iii) $S(0)$ is the identity on C;

(iv) $S(t)\big(S(\tau)x\big) = S(t+\tau)x$ for all $x \in C$ and $t, \tau \geq 0$.

It is important to note that all the conditions in Definition 20 are identical to conditions in Definition 16 with additional *strong continuity* condition given by Definition 20-*(ii)*. Therefore, in the case of *semigroup of bounded linear operators*, dynamical system given by Definition 20 reduces to C_0-semigroup definition.

Definition 21 (Equilibrium Point, Definition 4.1.2) *Let $\{S(t),\ t \geq 0\}$ be a dynamical system on C and for any $x \in C$, let $\Upsilon(x) = \{S(t)x,\ t \geq 0\}$ be the orbit through x. We say x is an equilibrium point if $\Upsilon(x) = \{x\}$. $\Upsilon(x)$ is a periodic orbit if there exists $p > 0$ such that $\Upsilon(x) = \{S(t)x,\ p \geq t \geq 0\} \neq \{x\}$.*

Definition 22 (Stability, Definition 4.1.2) *An orbit $\Upsilon(x)$ (or sometimes, the point x) is*

- *stable if $S(t)y \to S(t)x$ as $y \to x,\ y \in C$, uniformly in $t \geq 0$; i.e. if for any $\epsilon > 0$ there exists $\delta(\epsilon) > 0$ such that for all $t \geq 0$, $\|S(t)y - S(t)x\|_C < \epsilon$ whenever $\|y - x\|_C < \delta(\epsilon)$, $y \in C$,*

- *unstable if it is not stable,*

- *uniformly asymptotically stable if it is stable and also there is a neighborhood $\mathcal{N} = \{y \in C : \|y - x\|_C < r\}$ such that $\|S(t)y - S(t)x\|_C \to 0$ as $t \to \infty$, uniformly for $y \in \mathcal{N}$,*

- *exponentially asymptotically stable if there exists $\alpha, \beta > 0$ such that $\|S(t)y - S(t)x\|_C \leq \alpha \|y - x\|_C\ e^{-\beta(t-t_0)}$, for all $t \geq t_0$ and all $x, y \in C$,*

where $\|\cdot\|_C$ is the norm defined on C.

Definition 23 (Lyapunov function, Definition 4.1.3) *Let $\{S(t),\ t \geq 0\}$ be a dynamical system on C. A Lyapunov function is a continuous real-valued function V on C such that,*

$$\dot{V}(x) = \limsup_{t \to 0^+} \frac{V\big(S(t)x\big) - V(x)}{t} \leq 0$$

for all $x \in C$. Note that the possibility of $\dot{V}(x)$ being $-\infty$ is not excluded.

Theorem 17 (Lyapunov Stability, Theorem 4.1.4) *Let $\{S(t),\ t \geq 0\}$ be a dynamical system on C, and let 0 be an equilibrium point in C. Suppose V is a Lyapunov function on C which satisfies $V(0) = 0$, $V(x) \geq c(\|x\|)$ for $x \in C$, where $c(\cdot)$ is a continuous strictly increasing function, $c(0) = 0$, $c(r) > 0$ for $r > 0$. Then 0 is stable.*

Suppose in addition that $\dot{V}(x) \leq -c_1(\|x\|)$, where $c_1(\cdot)$ is also continuous, increasing and positive, with $c_1(0) = 0$. Then 0 is uniformly asymptotically stable.

Definition 24 (Invariant Set, Definition 4.3.1) *Let $\{S(t),\ t \geq 0\}$ be a dynamical system on a complete metric space C. A set $K \subset C$ is invariant if, for any $x_0 \in K$, there exists a continuous curve $x : \mathbb{R} \to K$ with $x(0) = x_0$ and*

$$S(t)x(\tau) = x(t+\tau) \quad \text{for} \quad -\infty < \tau < \infty,\ t \geq 0.$$

Theorem 18 (LaSalle Invariance Principle, Theorem 4.3.4) *Let V by a Lyapunov function on C (so $\dot{V}(x) \leq 0$), define*

$$E = \{x \in C : \dot{V}(x) = 0\},$$

and let M be the largest invariant subset of E. If $\{S(t)x_0,\ t \geq 0\}$ lies in a compact set in C, then $S(t)x_0 \to M$ as $t \to \infty$.