UNIVERSITY OF CALIFORNIA
Los Angeles

A Blob Method for Advection-Diffusion-Reaction Systems with Application to Robotic Swarms

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mathematics

by

Fangbo Zhang

2018

ABSTRACT OF THE DISSERTATION

A Blob Method for Advection-Diffusion-Reaction Systems
with Application to Robotic Swarms

by

Fangbo Zhang
Doctor of Philosophy in Mathematics
University of California, Los Angeles, 2018
Professor Andrea L. Bertozzi, Chair

In this dissertation, a blob method of measuring spatial coverage by a swarm of agents is presented, and two models are introduced: a macroscopic model, consisting of a system of Advection-Diffusion-Reaction equations that govern the spatial distribution of the swarm at the population level; and a microscopic model, represented by a stochastic differential equation, describing the individual behavior of each agent in the swarm. Depending on different tasks, multiple control frameworks are proposed to drive the swarm to a target distribution, and are proved to be both valid in theory and robust in real world case.

First, we briefly review the history of Advection-Diffusion-Reaction (ADR) system in the literature. The ADR system models the agents' motion as drifted brownian motion, and the agents' reaction as probabilistic transition between different states. The key idea of the ADR model lies in the fact that the green's function of Advection-Diffusion Equation is the conditional probability density function of the corresponding drifted brownian motion. The proposed point mass approach verifies that the error between the macroscopic and the microscopic model converges with respect to the inverse of the square root of the swarm size.

Next, we explores a stochastic approach for controlling swarms of independent robots toward a target distribution in a bounded domain. The robots are resource-constraint: they lack both communication and localization capabilities, and can only gather local information by measuring a scalar field (e.g. concentration of a chemical) from the environment. A simple

control law is introduced to govern the diffusion of each robot, so that the distribution of the swarm converges to a pre-defined target distribution over time. The key point behind the control framework is that the solution of heat equation converges to the first-order eigenvector of the Laplacian operator. We further confirm the robustness of the control law in real world case by conducting simulations in computer and test bed experiments in multiple cases of target distributions, where the swarm achieves the theorized convergence to the target distribution despite deviations from assumptions underpinning the theory.

Finally, we presents a novel procedure for computing parameters of a robotic swarm that guarantee coverage performance by the swarm within a specified error from a target spatial distribution. The main contribution is the analysis of the dependence of this error on two key parameters: the number of robots in the swarm and the robot sensing radius, in which we view each robot as a blob instead of a point mass. We model the population dynamics of the swarm as an advection-diffusion-reaction partial differential equation (PDE) with time-dependent advection and reaction terms. We derive rigorous bounds on the discrepancies between the target distribution and the coverage achieved by individual-based and PDE models of the swarm. We use these bounds to select the swarm size that will achieve coverage performance within a given error and the corresponding robot sensing radius that will minimize this error. We also apply the optimal control approach to compute the robots' velocity field and task-switching rates. We validate our procedure through simulations of a scenario in which a robotic swarm must achieve a specified density of pollination activity over a crop field.

The dissertation of Fangbo Zhang is approved.

Inwon C. Kim

Spring M. Berman

Wotao Yin

Andrea L. Bertozzi, Committee Chair

University of California, Los Angeles

2018

To my family, mentors, friends, and collaborators . . .

who believe in me and give me guidance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank above all my advisor, Andrea Bertozzi, for her patient guidance and strong support. Andrea has been the most important person in my academic life. She is exceptionally knowledgeable in so many research areas, and always navigates me in the right direction. Andrea, thank you for enlightening me for the past few years, and your guidance and insights made possible the research that is presented in this dissertation.

I would also like to thank my other doctoral committee members, Inwon Kim, Spring Berman, and Wotao Yin, for their time and valuble feedback. Besides their suggestions and advice in this dissertation, I have also had the pleasure to learn from them through attending their lectures and reading their papers. They demonstrate the ideals of top scientific researchers that I will strive to become. I am also grateful to the instructors of the courses that I have taken during my graduate study at UCLA. The wisdom they have imparted upon me has helped me develop a solid foundation in mathematics for future research.

I express my deep gratitude to two of my great friends and collaborators, Karthik Elamvazhuthi and Matt Haberland, who are both extremely brilliant and diligent researchers in their own fields. The valuable insights and advice they provided have been so important in my graduate study and research. As a graduate mentor, I would also like to thank Hanjun Li, Chunhan Feng, Henry Ehrhard, Yijun Shen and Bernardo Cobos for the excellent work in the REU program of summer 2016. Surrounding myself with hard-working and brilliant researchers like them has helped shape me into a mathematician today.

The work in this dissertation was supported by NSF grants CMMI-1435709, CMMI-1436960, and DMS-1045536. I was also supported by the Girsky Fellowship Award during the PhD program.

VITA

2013 B.S. Mathematics, Shanghai Jiao Tong University

2013 - 2016 Graduate Teaching Assistant, UCLA

2013 - 2018 Graduate Research Assistant, UCLA

2015 Summer Undergraduate Research Project Mentor, UCLA REU

2016 Summer Undergraduate Research Project Mentor, UCLA REU

2016 Summer Industrial Research Project Mentor, IPAM RIPS

2017 Summer Software Engineer Intern, Facebook Inc.

2018 (Expected) Ph.D. Mathematics, University of California, Los Angeles

PUBLICATIONS

Performance Bounds On Spatial Coverage Task by Stochastic Robotic Swarms by F. Zhang, A. L. Bertozzi, K. Elamvazhuthi and S. Berman; IEEE Transaction of Automatic Control, vol. 63, no. 6, 2018, pp. 1473?1488.

Decentralized Stochastic Control of Robotic Swarm Density: Theory, Simulation, and Experiment by H. Li, C. Feng, H. Ehrhard, Y. Shen, B. Cobos, F. Zhang, K. Elamvazhuthi, S. Berman, M. Haberland, and A. L. Bertozzi; 2017 IEEE RSJ International Conference on Intelligent Robots and Systems.

CHAPTER 1

History and Application: The Advection-Diffusion-Reaction System

1.1 Introduction

In recent years, the Advection-Diffusion-Reaction (ADR) System has been widely applied to model the motion and stochastic transitions in a swarm of agents in nanomedicine [HBN13], coverage and field estimation [EAB16, LFE17], unknown environment mapping [REB18] and robotic pollination [EB15, BNH11]. A swarm system is often comprised of hundreds or thousands of inexpensive, relatively expendable agents that perform tasks on large spatial and temporal scale. They should work quickly, robustly, adaptively, and autonomously [EB15]. Recent advances in computing, sensing, actuation, power, control, and 3D printing technologies have boosted the production of light robots, making research and experiments in robotic swarm more popular. Tasks for robotic swarms include exploration, mapping, environmental monitoring, chemical source localization, search-and-rescue, surveillance, and reconnaissance.

The simpler Advection-Diffusion equation, or Fokker Planck Equation, has been comprehensively studied in the literature. The Monte Carlo method is a popular way to simulate the solution of the Fokker Planck equation [MWW07, CJL06], in which the key idea is to deploy a swarm of independent particles, whose trajectories are generated by the corresponding drifted Brownian motion of the Fokker Planck equation.

This chapter is organized as following. In section 1.2, we briefly review and prove the theory of the Fokker Planck equation, and present a numerical simulation using a point mass

approach. In section 1.3, we include the reaction term and show some examples of existing applications of ADR system in the literature.

1.2 Preliminary: Fokker-Planck Equation

In this section, we state and prove Kolmogorov theorem [Pav14], which demonstrates the fact that the Green's function of the Advection-Diffusion equation is the conditional probability density of a drifted Brownian motion.

Definition 1.2.1. *(Fokker-Planck) Suppose* $\boldsymbol{v}(\boldsymbol{x},t), D(\boldsymbol{x},t)$ *are functions in* $C^{2,1}(R^2, R_+)$*,* $D(\boldsymbol{x},t) > 0$*, and* $\boldsymbol{X}(t)$ *is a stochastic process that satisfies Stochastic Differential Equation(SDE)*

$$\boldsymbol{dX}(t) = \boldsymbol{v}(\boldsymbol{X}(t),t)dt + \sqrt{2D(\boldsymbol{X}(t),t)}\boldsymbol{dB}(t), \tag{1.1}$$

then the Fokker-Planck equation for the probability density $p(\boldsymbol{x},t)$ *of the random variable* $\boldsymbol{X}(t)$ *is*

$$\rho_t + \nabla \cdot (\boldsymbol{v}\rho) = \Delta(D\rho). \tag{1.2}$$

Theorem 1.2.2 (Kolmogorov theorem)**.** *Let* $p(\boldsymbol{y},t|\boldsymbol{x},s)$ *is the conditional probability density, such that* $\forall \Sigma \in \mathbb{R}^2, t > s > 0$*,*

$$P\{X(t) \in \Sigma | X(s) = \boldsymbol{x}\} = \int_\Sigma p(\boldsymbol{y},t|\boldsymbol{x},s)\boldsymbol{dy}.$$

Then $p(\boldsymbol{y},t|\boldsymbol{x},s)$ *is the solution to the initial value problem*

$$\begin{cases} p_t = -\nabla_{\boldsymbol{y}} \cdot (\boldsymbol{v}(\boldsymbol{y},t)p) + \Delta_{\boldsymbol{y}}(D(\boldsymbol{y},t)p), \ \forall \boldsymbol{y} \in \Omega \\ p(\boldsymbol{y},s|\boldsymbol{x},s) = \delta(\boldsymbol{y}-\boldsymbol{x}), \end{cases} \tag{1.3}$$

with the no-flux boundary condition

$$(-\boldsymbol{v}(\boldsymbol{y},t)p + \nabla_{\boldsymbol{y}}(D(\boldsymbol{y},t)p)) \cdot \overrightarrow{n} = 0, \ \forall \boldsymbol{y} \in \partial\Omega. \tag{1.4}$$

or the vanishing boundary condition in the case when $\Omega = \mathbb{R}^2$*:*

$$\lim_{\|\boldsymbol{x}\| \to \infty} p(\boldsymbol{x}) = 0. \tag{1.5}$$

Or equivalently, the conditional probability density function is the Green's function of Fokker-Planck Equation.

We will prove theorem 1.2.2 later in section 1.3. Based on the theorem, we obtain the following corollaries.

Corollary 1.2.3. *Let $p(\boldsymbol{y}, t|\boldsymbol{x}, s)$ is the conditional transaction probability density of the stochastic process defined in* (1.1)*, then the solution to the initial value problem*

$$\begin{cases} \rho_t = -\nabla \cdot (\boldsymbol{v}(\boldsymbol{x}, t)\rho) + \Delta(D(\boldsymbol{x}, t)\rho) \\ \rho(\boldsymbol{x}, 0) = \rho_0(\boldsymbol{x}), \end{cases} \tag{1.6}$$

with the same boundary condition in theorem 1.2.2 is:

$$\rho(\boldsymbol{x}, t) = \int_\Omega \rho_0(\boldsymbol{y}) p(\boldsymbol{x}, t|\boldsymbol{y}, 0) \boldsymbol{dy}.$$

1.2.1 First Simulation: Point Mass Approach

One well known way to simulate the distribution of drifted Brownian motion is the Monte Carlo method, in which one often view each particle as a point mass. We now suppose there are $N \in Z^+$ particles that are moving according to SDE (1.1) in a 2D plane, indexed by $1 \leq i \leq N$. The i^{th} particle starts from $\boldsymbol{X}_0^i \in \Omega$. For a fixed area $\Sigma \subset \Omega$, the total number of particles in Σ at time t is

$$\#\text{ of particles in } \Sigma = \sum_i \mathbb{1}_\Sigma(\boldsymbol{X}_t^i),$$

where $\mathbb{1}_\Sigma(\boldsymbol{x})$ is 1 if $\boldsymbol{x} \in \Sigma$ and 0 otherwise. Let $\delta(\boldsymbol{x})$ denote the delta function in 2D. Then if we see each particle as a point mass, the corresponding density function of the initial state of the particle system is

$$\rho_0(\boldsymbol{x}) = \sum_i \delta(\boldsymbol{x} - \boldsymbol{X}_0^i).$$

The total mass of the system is conserved since

$$\frac{\partial}{\partial t} \int_\Omega \rho \boldsymbol{dx} = \int_\Omega -\nabla \cdot (\boldsymbol{v}(\boldsymbol{x}, t)\rho) + \Delta(D(\boldsymbol{x}, t)\rho) \boldsymbol{dx} = \int_{\partial\Omega} (-\boldsymbol{v}p + \nabla(Dp)) \cdot \overrightarrow{n} \boldsymbol{dS} = 0.$$

Theorem 1.2.2 also leads to the following corollary.

Corollary 1.2.4. *Let $\rho(\boldsymbol{x}, t)$ be the solution to the initial value problem*

$$\begin{cases} \rho_t = -\nabla \cdot (\boldsymbol{v}(\boldsymbol{x}, t)\rho) + \Delta(D(\boldsymbol{x}, t)\rho), \ \forall \boldsymbol{x} \in \Omega \\ \rho_0(\boldsymbol{x}) = \dfrac{1}{N}\sum_i \delta(\boldsymbol{x} - \boldsymbol{X}_0^i), \end{cases} \tag{1.7}$$

with the same boundary condition in theorem 1.2.2. Without loss of generality we assume

$$\int_{\mathbb{R}^2} \rho(\boldsymbol{x}, t)\boldsymbol{dx} \equiv 1.$$

Then for any $\Sigma \subset \Omega$ and $t > 0$,

$$E(\textit{portion of particles in } \Sigma \textit{ at time } t) = \frac{1}{N}\sum_i E\left(\mathbb{1}_\Sigma(\boldsymbol{X}_t^i)\right) = \int_\Sigma \rho(\boldsymbol{x}, t)\boldsymbol{dx}. \tag{1.8}$$

Proof. By corollary 1.2.3, $\rho(\boldsymbol{x}, t) = \int_\Omega \rho_0(\boldsymbol{y})p(\boldsymbol{x}, t|\boldsymbol{y}, 0)\boldsymbol{dy}$, where $p(\boldsymbol{x}, t|\boldsymbol{y}, 0)$ is the conditional transaction probability density of (1.1). Denote $\boldsymbol{X}(t; \boldsymbol{x})$ be the stochastic process of (1.1) starting from $\boldsymbol{x}$, then we have

$$\begin{aligned} \int_\Sigma \rho(\boldsymbol{x}, t)\boldsymbol{dx} &= \int_\Omega \mathbb{1}_\Sigma(\boldsymbol{x})\rho(\boldsymbol{x}, t)\boldsymbol{dx} \\ &= \int_\Omega \mathbb{1}_\Sigma(\boldsymbol{x}) \int_\Omega \rho_0(\boldsymbol{y})p(\boldsymbol{x}, t|\boldsymbol{y}, 0)\boldsymbol{dy}\boldsymbol{dx} \\ &= \int_\Omega \rho_0(\boldsymbol{y}) \int_\Omega \mathbb{1}_\Sigma(\boldsymbol{x})p(\boldsymbol{x}, t|\boldsymbol{y}, 0)\boldsymbol{dx}\boldsymbol{dy} \\ &= \int_\Omega \rho_0(\boldsymbol{y})E\left(\mathbb{1}_\Sigma(\boldsymbol{X}(t; \boldsymbol{y}))\right)\boldsymbol{dy}. \end{aligned}$$

Now we plug in $\rho_0(\boldsymbol{y}) = \frac{1}{N}\sum_i \delta(\boldsymbol{y} - \boldsymbol{X}_0^i)$, and get

$$\int_\Sigma \rho(\boldsymbol{x}, t)\boldsymbol{dx} = \frac{1}{N}\sum_i E\left(\mathbb{1}_\Sigma\left(\boldsymbol{X}(t; \boldsymbol{X}_0^i)\right)\right) = \frac{1}{N}\sum_i E\left(\mathbb{1}_\Sigma(\boldsymbol{X}_t^i)\right).$$

□

1.2.2 Numerical Experiment

We present an numerical simulation that justifies corollary 1.2.3. We choose $N = 10000, \boldsymbol{v} = (0.5x_1, 0.5x_2), D = 1$, $\Sigma = [0, 1] \times [0.2, 1.2] \subset \mathbb{R}^2$, all of the particles start from origin. We discretize the time interval $[0, 1]$ into $M = 1000$ equal time steps as

$$0 = t_0 < t_1 < ... < t_M = T, \quad t_m = m\Delta t. \tag{1.9}$$

We simulate each particle path independently by

$$\boldsymbol{X}_{m+1}^i = \boldsymbol{X}_m^i + \boldsymbol{v}(t_m)\Delta t + \sqrt{2D\Delta t}\Delta \boldsymbol{Z}_m^i \quad \forall i \in F_{m+1}, \tag{1.10}$$

where $\boldsymbol{X}_m^i$ denote the position of i^{th} particle at m^{th} time step, and $\Delta \boldsymbol{Z}_m^i$ are independent, normally distributed random variables with zero mean and unit variance in $\mathbb{R}^2$. Figure 1.1 shows final distribution of a sampled 500 particles, and the comparison between portion of particles in Σ and $\int_\Sigma \rho(\boldsymbol{x}, t)\boldsymbol{dx}$.

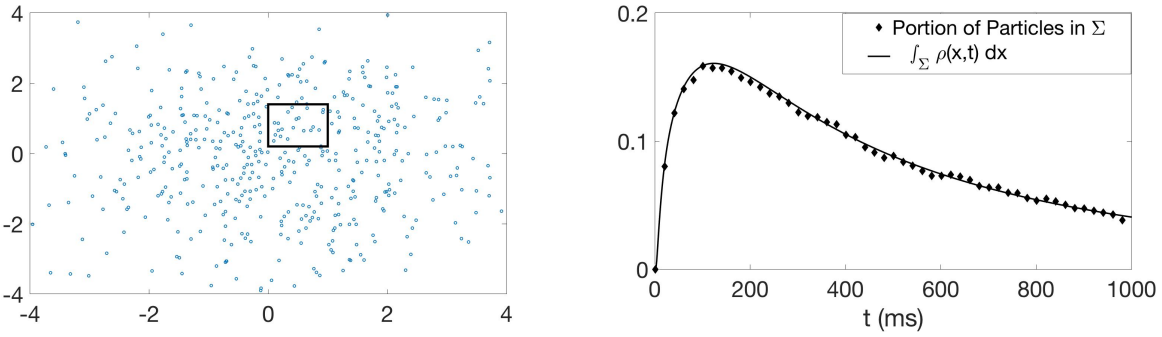


Figure 1.1: Left: Final distribution of sampled 500 particles, where Σ is depicted as the solid rectangle. Right: portion of particles in $\Sigma = [0, 1] \times [0.2, 1.2]$ (diamond) and $\int_\Sigma \rho(\boldsymbol{x}, t)\boldsymbol{dx}$ (solid line) over time.

Also we define the error

$$e(N) = \int_\Sigma \rho(\boldsymbol{x}, T)\boldsymbol{dx} - \frac{1}{N}\sum_i \mathbb{1}_\Sigma(\boldsymbol{X}_T^i)$$

. From central limit theorem we know that

$$\sqrt{N}e(N) = \frac{1}{\sqrt{N}}\sum_i \left(\mathbb{1}_\Sigma(\boldsymbol{X}_T^i) - E\left(\mathbb{1}_\Sigma(\boldsymbol{X}_T^i)\right)\right) \sim \mathcal{N}(0,\, \sigma^2)$$

Hence we conjecture that $e(N) \sim \frac{1}{\sqrt{N}}$. In figure 1.2, $\ln e(N)$ is computed against $\ln N$, where the results are the average of 100 experiments. We discover linear dependency between them, which will be rigorously proved later in a more complex setting.

1.2.3 Proof of Kolmogorov Theorem

In this subsection, we present the proof of the Kolmogorov theorem 1.2.2. A proof for a 1D diffusion process can be found in the book [Pav14], Section 2.5, and the proof here is a

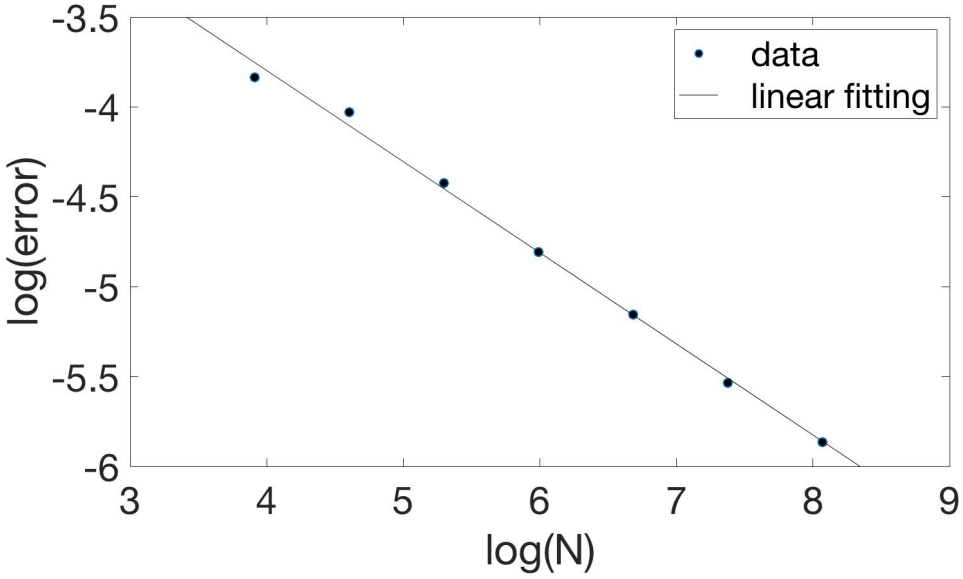


Figure 1.2: Different $e(N)$ with respect to different N, all the results are average of 100 experiments.

generalization in the 2D case. To simplify the proof, we consider the case when $\Omega = \mathbb{R}^2$. We first state two lemmas.

Lemma 1.2.5. *With the stochastic process $\boldsymbol{X}(t) = (X_1(t), X_2(t))$ defined above, we have:*
(i)First moment:

$$E\left(\boldsymbol{X}(t) - \boldsymbol{X}(s)\middle|\boldsymbol{X}(s) = \boldsymbol{x}\right) = \boldsymbol{v}(\boldsymbol{x}, s)(t-s) + o(t-s) \tag{1.11}$$

(ii)Second moment:

$$\begin{aligned} &E\left((X_i(t) - X_i(s))(X_j(t) - X_j(s))\middle|\boldsymbol{X}(s) = \boldsymbol{x}\right) \\ &= 2D(\boldsymbol{x}, s)(t-s)\delta_{ij} + o(t-s) \end{aligned} \tag{1.12}$$

(iii)Higher order moment:

$$\forall \delta > 0,\ E\left(|\boldsymbol{X}(t) - \boldsymbol{X}(s)|^{2+\delta}\right) = o(t-s) \tag{1.13}$$

where $i, j \in \{1, 2\}$, δ_{ij} is 1 when $i = j$, or 0 otherwise.

Lemma 1.2.6. *With the process $\boldsymbol{X}(t)$ defined above, for every fixed $\boldsymbol{x}$ and every $\epsilon > 0$:*

(i)

$$\int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} = o(t-s) \tag{1.14}$$

uniformly over $s < t$.

(ii)

$$\int_{|\boldsymbol{y}-\boldsymbol{x}|\leq\epsilon} (\boldsymbol{y}-\boldsymbol{x})p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} = \boldsymbol{v}(\boldsymbol{x},s)(t-s) + o(t-s) \tag{1.15}$$

uniformly over $s < t$.

(iii) Write $\boldsymbol{x} = (x_1, x_2), \boldsymbol{y} = (y_1, y_2)$, *then*

$$\int_{|\boldsymbol{y}-\boldsymbol{x}|\leq\epsilon} (y_i - x_i)(y_j - x_j)p(\boldsymbol{y},t|\boldsymbol{x},s)dy = 2D(\boldsymbol{x},s)(t-s)\delta_{ij} + o(t-s) \tag{1.16}$$

uniformly over $s < t$.

We postpone the proof of the above two lemmas, and apply them to prove our main theorem first.

Proof of Theorem 1.2.2. The initial condition follows from the definition. Now for any function $f(\boldsymbol{y}) \in C_0^2(R^2)$, notice that $p(\boldsymbol{y},t|\boldsymbol{x},s)$ satisfies Chapman-Kolmogorov equation:

$$p(\boldsymbol{y},t+k|\boldsymbol{x},s) = \int_{R^2} p(\boldsymbol{y},t+k|\boldsymbol{z},t)p(\boldsymbol{z},t|\boldsymbol{x},s)d\boldsymbol{z}. \tag{1.17}$$

Hence

$$\begin{aligned}
&\int_{R^2} f(\boldsymbol{y})\frac{\partial}{\partial t}p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
=&\lim_{k\to 0}\frac{1}{k}\left(\int_{R^2} f(\boldsymbol{y})p(\boldsymbol{y},t+k|\boldsymbol{x},s)d\boldsymbol{y} - \int_{R^2} f(\boldsymbol{y})p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y}\right) \\
=&\lim_{k\to 0}\frac{1}{k}\left(\int_{R^2}\int_{R^2} f(\boldsymbol{y})p(\boldsymbol{y},t+k|\boldsymbol{z},t)p(\boldsymbol{z},t|\boldsymbol{x},s)d\boldsymbol{y}d\boldsymbol{z} - \int_{R^2} f(\boldsymbol{z})p(\boldsymbol{z},t|\boldsymbol{x},s)d\boldsymbol{z}\right) \\
=&\lim_{k\to 0}\frac{1}{k}\left(\int_{R^2} p(\boldsymbol{z},t|\boldsymbol{x},s)\left(\int_{R^2} f(\boldsymbol{y})p(\boldsymbol{y},t+k|\boldsymbol{z},t)d\boldsymbol{y} - f(\boldsymbol{z})\right)d\boldsymbol{z}\right).
\end{aligned} \tag{1.18}$$

Let $Hf(\boldsymbol{z})$ denote the 2×2 Hessian matrix of f at $\boldsymbol{z}$, then $\forall \epsilon$ we have

$$
\begin{aligned}
&\int_{R^2} f(\boldsymbol{y}) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} - f(\boldsymbol{z}) \\
=&\int_{R^2} (f(\boldsymbol{y}) - f(\boldsymbol{z})) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} \\
=&\int_{|\boldsymbol{y}-\boldsymbol{z}|\leq\epsilon} (f(\boldsymbol{y}) - f(\boldsymbol{z})) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} + \int_{|\boldsymbol{y}-\boldsymbol{z}|>\epsilon} (f(\boldsymbol{y}) - f(\boldsymbol{z})) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} \\
\leq&\int_{|\boldsymbol{y}-\boldsymbol{z}|\leq\epsilon} (f(\boldsymbol{y}) - f(\boldsymbol{z})) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} + 2\|f\|_\infty o(k) \\
=&\int_{|\boldsymbol{y}-\boldsymbol{z}|\leq\epsilon} \left(\nabla f(\boldsymbol{z})(\boldsymbol{y}-\boldsymbol{z}) + \frac{1}{2}(\boldsymbol{y}-\boldsymbol{z})^T Hf(\boldsymbol{z})(\boldsymbol{y}-\boldsymbol{z})(1+O(|\boldsymbol{y}-\boldsymbol{z}|)) \right) \\
&\qquad\qquad \cdot p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} + o(k) \\
=&\nabla f(\boldsymbol{z}) \boldsymbol{v}(\boldsymbol{z}, t) k + D(\boldsymbol{z}, t) k \Delta f(\boldsymbol{z})(1+O(\epsilon)) + o(k),
\end{aligned}
$$

where we applied (1.11) and (1.12) in the last step. Taking $\epsilon \to 0$ we get

$$
\int_{R^2} f(\boldsymbol{y}) p(\boldsymbol{y}, t+k|\boldsymbol{z}, t) d\boldsymbol{y} - f(\boldsymbol{z}) = k \left(\nabla f(\boldsymbol{z}) \boldsymbol{v}(\boldsymbol{z}, t) + D(\boldsymbol{z}, t) \Delta f(\boldsymbol{z}) \right) + o(k). \tag{1.19}
$$

Now plug (1.19) back into (1.18), use 2D integration by parts formula, and apply the boundary condition, we get:

$$
\begin{aligned}
&\int_{R^2} f(\boldsymbol{y}) \frac{\partial}{\partial t} p(\boldsymbol{y}, t|\boldsymbol{x}, s) d\boldsymbol{y} \\
=&\int_{R^2} p(\boldsymbol{z}, t|\boldsymbol{x}, s) \left(\nabla f(\boldsymbol{z}) \boldsymbol{v}(\boldsymbol{z}, t) + D(\boldsymbol{z}, t) \Delta f(\boldsymbol{z}) \right) d\boldsymbol{z} \\
=&\int_{R^2} f(\boldsymbol{y}) \left(-\nabla \cdot (\boldsymbol{v}(\boldsymbol{y}, t) p + \Delta (D(\boldsymbol{y}, t) p) \right) d\boldsymbol{y}
\end{aligned}
$$

Since $f(\boldsymbol{y})$ is arbitrarily chosen in $C_0^2(R^2)$, we know that

$$
p_t = -\nabla_y \cdot (\boldsymbol{v}(\boldsymbol{y}, t) p) + \frac{1}{2} \Delta_y (D(\boldsymbol{y}, t) p).
$$

This closes the proof of the theorem. □

Now what is left is to prove the lemma 1.2.5 and lemma 1.2.6.

Proof of lemma 1.2.5. From SDE (1.1) we know that

$$
\boldsymbol{X}(t) - \boldsymbol{X}(s) = \int_s^t \boldsymbol{v}(\boldsymbol{X}(\tau), \tau) d\tau + \int_s^t \sqrt{2D(\boldsymbol{X}(\tau), \tau)} \boldsymbol{dB}(\tau) \tag{1.20}
$$

So, by property of conditional expectation,

$$\begin{aligned}
&E\bigg(\boldsymbol{X}(t)-\boldsymbol{X}(s)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)-\boldsymbol{v}(\boldsymbol{x},s)(t-s)\\
=&E\bigg(\boldsymbol{X}(t)-\boldsymbol{X}(s)-\boldsymbol{v}(\boldsymbol{X}(s),s)(t-s)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg).
\end{aligned}$$

Notice that

$$E\bigg(\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}d\boldsymbol{B}(\tau)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)=0,$$

so the above identity becomes

$$\begin{aligned}
&E\bigg(\boldsymbol{X}(t)-\boldsymbol{X}(s)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)-\boldsymbol{v}(\boldsymbol{x},s)(t-s)\\
=&E\bigg(\int_s^t\boldsymbol{v}(\boldsymbol{X}(\tau),\tau)d\tau-\boldsymbol{v}(\boldsymbol{X}(s),s)(t-s)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)\\
=&E\bigg(\int_s^t\boldsymbol{v}(\boldsymbol{X}(\tau),\tau)-\boldsymbol{v}(\boldsymbol{X}(s),s)d\tau\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)\\
\leq&E\bigg(\int_s^t|\boldsymbol{v}(\boldsymbol{X}(\tau),\tau)-\boldsymbol{v}(\boldsymbol{X}(s),s)|d\tau\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)
\end{aligned}$$

But by continuity of $\boldsymbol{v}(\boldsymbol{x},t)$ and almost everywhere continuity of $\boldsymbol{X}(t)$,

$$\int_s^t|\boldsymbol{v}(\boldsymbol{X}(\tau),\tau)-\boldsymbol{v}(\boldsymbol{X}(s),s)|d\tau=\int_s^tO(t-s)d\tau=o(t-s).$$

Hence

$$E\bigg(\boldsymbol{X}(t)-\boldsymbol{X}(s)\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)-\boldsymbol{v}(\boldsymbol{x},s)(t-s)=o(t-s),$$

which proves (i). For (ii), consider the case $i\neq j$ first, if we write $\boldsymbol{B}(t)=(B_1(t),B_2(t)),\boldsymbol{v}=(v_1,v_2)$, then SDE (1.1) becomes:

$$\begin{aligned}
dX_1(t)&=v_1(\boldsymbol{X}(t),t)dt+\sqrt{2D(\boldsymbol{x},t)}dB_1(t)\\
dX_2(t)&=v_2(\boldsymbol{X}(t),t)dt+\sqrt{2D(\boldsymbol{x},t)}dB_2(t).
\end{aligned}$$

Since the stochastic terms $B_1(t)$ and $B_2(t)$ are independent of each other, so do $X_1(t)$ and

$X_2(t)$. Hence

$$\begin{aligned}
&E\bigg((X_1(t)-X_1(s))(X_2(t)-X_2(s))\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)\\
=&E\bigg((X_1(t)-X_1(s))\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)E\bigg((X_2(t)-X_2(s))\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)\\
=&(v_1(\boldsymbol{X}(s),s)(t-s)+o(t-s))\cdot(v_2(\boldsymbol{X}(s),s)(t-s)+o(t-s))\\
=&o(t-s).
\end{aligned}$$

Now suppose $i=j$, WLOG $i=j=1$. $\boldsymbol{B}(t)$ has independent increment, hence $X_1(t)-X_1(s)$ and $\boldsymbol{X}(s)$ are independent.

$$\begin{aligned}
&E\bigg((X_1(t)-X_1(s))^2\bigg|\boldsymbol{X}(s)=\boldsymbol{x}\bigg)\\
=&E\bigg((X_1(t)-X_1(s))^2\bigg)\\
=&E\bigg(\bigg[\int_s^t v_1(\boldsymbol{X}(\tau),\tau)d\tau+\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg]^2\bigg)
\end{aligned}$$

with the consumption that $\boldsymbol{X}(s)=\boldsymbol{x}$. Notice that

$$\begin{aligned}
&E\bigg(\bigg[\int_s^t v_1(\boldsymbol{X}(\tau),\tau)d\tau\bigg]^2\bigg)=E\bigg((O(t-s))^2\bigg)=o(t-s);\\
&E\bigg(\bigg[\int_s^t v_1(\boldsymbol{X}(\tau),\tau)d\tau\bigg]\cdot\bigg[\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg]\bigg)\\
&\le E\bigg(\bigg|\int_s^t v_1(\boldsymbol{X}(\tau),\tau)d\tau\bigg|\cdot\bigg|\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg|\bigg)\\
&\le E\bigg(C(t-s)\bigg|\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg|\bigg)\\
&\le C(t-s)E\bigg(\bigg|\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg|^2\bigg)^{\frac{1}{2}}\\
&=C(t-s)\cdot E\bigg(\int_s^t 2D(\boldsymbol{X}(\tau),\tau)d\tau\bigg)^{\frac{1}{2}}\\
&\le C'(t-s)^{\frac{3}{2}}=o(t-s);\\
&E\bigg(\bigg[\int_s^t\sqrt{2D(\boldsymbol{X}(\tau),\tau)}dB_1(\tau)\bigg]^2\bigg)=E\bigg(\int_s^t 2D(\boldsymbol{X}(\tau),\tau)d\tau\bigg)
\end{aligned}$$

with the consumption that $\boldsymbol{X}(s) = \boldsymbol{x}$. Hence we have

$$\begin{aligned}
&E\bigg((X_1(t) - X_1(s))^2 \Big| \boldsymbol{X}(s) = \boldsymbol{x}\bigg) - 2D(\boldsymbol{x}, s)(t-s) \\
=&E\bigg(\int_s^t 2D(\boldsymbol{X}(\tau), \tau)d\tau\bigg) - 2D(\boldsymbol{x}, s)(t-s) + o(t-s) \\
=&E\bigg(\int_s^t 2D(\boldsymbol{X}(\tau), \tau) - 2D(\boldsymbol{X}(s), s)d\tau\bigg) + o(t-s) \\
=&E\bigg(\int_s^t O(t-s)d\tau\bigg) + o(t-s) \\
=&o(t-s),
\end{aligned}$$

which completes the proof of (ii).

For (iii), we use the inequality $(a^2+b^2)^{\frac{2+\delta}{2}} \leq 2^{\frac{2+\delta}{2}}(a^{2+\delta} + b^{2+\delta})$, so that it is sufficient to prove

$$E\bigg(|X_1(t) - X_1(s)|^{2+\delta}\bigg) = o(t-s).$$

By (1.20), and use the above inequality again, we get

$$\begin{aligned}
&E\bigg(|X_1(t) - X_1(s)|^{2+\delta}\bigg) = E\bigg(\bigg|-\int_s^t v_1(\boldsymbol{X}(\tau), \tau)d\tau + \int_s^t \sqrt{2D(\boldsymbol{X}(\tau), \tau)}dB_1(\tau)\bigg|^{2+\delta}\bigg) \\
&\leq 2^{\frac{2+\delta}{2}}\left[E\bigg(\bigg|\int_s^t v_1(\boldsymbol{X}(\tau), \tau)d\tau\bigg|^{2+\delta}\bigg) + E\bigg(\bigg|\int_s^t \sqrt{2D(\boldsymbol{X}(\tau), \tau)}dB_1(\tau)\bigg|^{2+\delta}\bigg)\right].
\end{aligned}$$

The first term is $O(|t-s|^{2+\delta})$ since v_1 is bounded. For the second term, we apply Holder's inequality $E(|X|^{\delta+2}) \leq E(|X|^2)^{(\delta+2)/2}$, and get

$$\begin{aligned}
&E\bigg(\bigg|\int_s^t \sqrt{2D(\boldsymbol{X}(\tau), \tau)}dB_1(\tau)\bigg|^{2+\delta}\bigg) \\
&\leq \left[E\bigg(\bigg|\int_s^t \sqrt{2D(\boldsymbol{X}(\tau), \tau)}dB_1(\tau)\bigg|^{2}\bigg)\right]^{(\delta+2)/2} \\
&= \left[E\bigg(\int_s^t 2D(\boldsymbol{X}(\tau), \tau)d\tau\bigg)\right]^{(\delta+2)/2} = |t-s|^{1+\delta/2} = o(t-s).
\end{aligned}$$

which closes the proof. □

Proof of lemma 1.2.6. Equation (1.13) is equivalent to:

$$\int_{R^2} |\boldsymbol{y} - \boldsymbol{x}|^{2+\delta} p(\boldsymbol{y}, t|\boldsymbol{x}, s)d\boldsymbol{y} = o(t-s), \forall \delta > 0$$

So for every $\epsilon > 0$ and take r to be 0, 1 or 2,

$$\begin{aligned}
&\int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} |\boldsymbol{y}-\boldsymbol{x}|^r p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
=&\int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} \frac{|\boldsymbol{y}-\boldsymbol{x}|^{2+\delta}}{|\boldsymbol{y}-\boldsymbol{x}|^{2+\delta-r}} p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
\leq&\frac{1}{\epsilon^{2+\delta-r}} \int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} |\boldsymbol{y}-\boldsymbol{x}|^{2+\delta} p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
\leq&\frac{1}{\epsilon^{2+\delta-r}} \int_{R^2} |\boldsymbol{y}-\boldsymbol{x}|^{2+\delta} p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
=&o(t-s).
\end{aligned}$$

Taking $r = 0$ gives us (i). Also Equation (1.11) tells us that

$$\begin{aligned}
&\boldsymbol{v}(\boldsymbol{x},s)(t-s) + o(t-s) \\
=&\int_{R^2} (\boldsymbol{y}-\boldsymbol{x}) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
=&\int_{|\boldsymbol{y}-\boldsymbol{x}|\leq\epsilon} (\boldsymbol{y}-\boldsymbol{x}) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} + \int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} (\boldsymbol{y}-\boldsymbol{x}) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y}.
\end{aligned}$$

The later term is $o(t-s)$ by taking $r = 1$, which proves (ii). For (iii), we write (1.12) as

$$\begin{aligned}
&2D(\boldsymbol{x},s)(t-s)\delta_{ij} + o(t-s) = \int_{R^2} (y_i - x_i)(y_j - x_j) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} \\
=&\int_{|\boldsymbol{y}-\boldsymbol{x}|\leq\epsilon} (y_i - x_i)(y_j - x_j) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y} + \int_{|\boldsymbol{y}-\boldsymbol{x}|>\epsilon} (y_i - x_i)(y_j - x_j) p(\boldsymbol{y},t|\boldsymbol{x},s)d\boldsymbol{y}.
\end{aligned}$$

Combining with inequality $|(y_i - x_i)(y_j - x_j)| \leq |\boldsymbol{y}-\boldsymbol{x}|^2$, the later term is $o(t-s)$. This proves (iii), thus closes the whole lemma. □

1.3 Advection-Diffusion-Reaction System, and its applications

Consider the following system of partial differential equations

$$\begin{cases}
\dfrac{\partial \rho_1}{\partial t} = -\boldsymbol{v}_1 \cdot \nabla \rho_1 + D_1 \Delta \rho_1 - k_{12}\rho_1 + k_{21}\rho_2, \\
\dfrac{\partial \rho_2}{\partial t} = -\boldsymbol{v}_2 \cdot \nabla \rho_2 + D_2 \Delta \rho_2 + k_{12}\rho_1 - k_{21}\rho_2.
\end{cases} \tag{1.21}$$

Compared to the Advection-Diffusion equation (1.2), the system (1.21) has particles in two states, namely state 1 and state 2, corresponding to densities ρ_1, ρ_2, respectively. Particles

in state $n(n = 1, 2)$ are moving with advection $\boldsymbol{v}_n$ and diffusing with coefficient D_n, and they also react with each other, i.e. particle in state 1 switches to state 2, and state 2 also switches to state 1, with transition rate k_{12}, k_{21}, respectively. We call system (1.21) an Advection-Diffusion-Reaction (ADR) system. A specific example for the application of the ADR system is in the field of nano-medicine [HBN13], where different species of nanoparticles react with each other, resulting in an interchange network between species.

In the rest of this section, we briefly present some applications of ADR system that have been studied in the literature.

1.3.1 Targeted nanoparticles in Cancer Treatment

In cancer treatment, targeted nanoparticles has been increasingly utilized. However, the penetration of targeted nanoparticles deep into tissue can be hindered by their slow diffusion. In order to increase tissue penetration, S. Huert et al. [HBN13] proposed generalizable ADR model that captures the potency, motion, and binding kinetics of four species of nanoparticles: NP_F, free nanoparticles; NP_I, internalized nanoparticles; R, receptors; and C, nanoparticle-receptor complexes. The reaction network is:

$$\begin{aligned} NP_F + R &\xrightarrow{k_a} C, \\ NP_F + R &\xleftarrow{k_d} C, \\ C &\xrightarrow{k_i} NP_I + R, \end{aligned} \tag{1.22}$$

where k_a, k_d, k_i are reaction rates. The corresponding ADR system, or *the deterministic model*, is defined as

$$\begin{cases} \dfrac{\partial NP_F}{\partial t} = D\Delta NP_F - k_a NP_F \cdot R + k_d C, \\ \dfrac{\partial R}{\partial t} = -k_a NP_F \cdot R + k_d C + k_i C, \\ \dfrac{\partial C}{\partial t} = k_a NP_F \cdot R - k_d C - k_i C, \\ \dfrac{\partial NP_I}{\partial t} = k_i C, \end{cases} \tag{1.23}$$

with boundary condition

$$\left.\frac{\partial NP_F}{\partial t}(x,t)\right|_{x\in\partial\Omega} = 0.$$

In this system, the diffusion term is $D\Delta NP_F$; the reaction terms are $k_a NP_F \cdot R, k_d C, k_i C$, corresponding to 3 types of reactions in (1.22); there is no advection. Hence this is a Diffusion-Reaction (DR) system. The movement of free nanoparticles NP_F is modeled as Brownian motion

$$\boldsymbol{dX}(t) = \sqrt{2D(\boldsymbol{X}(t), t)}\boldsymbol{dB}(t). \tag{1.24}$$

The combination of (1.22) and (1.24) is called *the stochastic model*.

In the simulation, the authors explore parameters that affect the distribution of nanoparticles after extravasation. Such particles can be modified through engineering. Toward this end, they define dissociation constant $k_D = k_d/k_a$. Fig. 1.3A shows the number of cells that are dead after 48 hours of treatment as a function of D and K_D, simulated using both the stochastic model and the deterministic model. Note that the identical results from both models validate the use of the deterministic model for prediction and optimization. Based on experimental work by Hong et al. [HLM07] and Thurber et al. [TW08], they estimated the dissociation and internalization rate constants as $k_d = 10^{-4}s^{-1}, k_i = 10^{-5}s^{-1}$ while varying k_a in the range $[103, 109]M^{-1}s^{-1}$. Fig. 1.3B shows the penetration profiles of several nanoparticle formulations. The results show that most of the nanoparticle formulations considered are not capable of killing all 20 cells in the model. Nanoparticles with a high binding affinity (Fig. 1.3B(a,c)), regardless of their speed, accumulate only in cells near the vasculature, and slow nanoparticles (Fig. 1.3B(c,d)) fail to accumulate at lethal levels in cells farthest from the vasculature. Fast nanoparticles with a low binding affinity are able to accumulate at lethal levels in all cells (Fig. 1.3B(b)). However, lowering the affinity beyond the range explored here could result in nanoparticles that are unable to accumulate at lethal levels in tumor cells.

1.3.2 Coverage and Field Estimation on Bounded Domain

In [EAB16], K. Elamvazhuthi el at. developed a generic control framework to achieve and estimate a certain coverage in a bounded domain by deploying a swarm of agents.

Consider the following background. A swarm of agents that are deployed into a domain

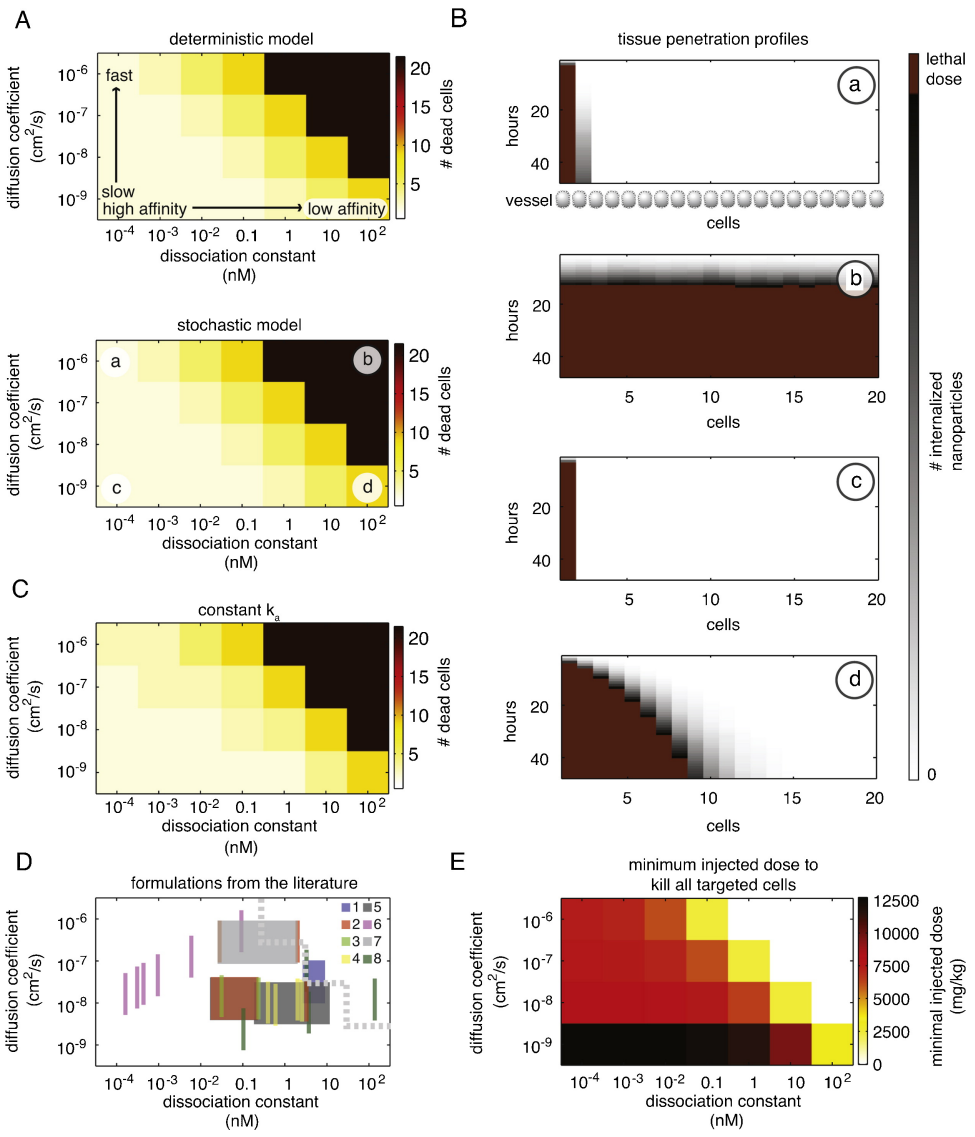


Figure 1.3: (A) Number of cells killed depending on the diffusion coefficient D and dissociation constant K_D. Complete tissue penetration is assumed when each of the 20 cells in the model internalizes the number of nanoparticles required to kill one cell. (B) Tissue penetration profiles determined using a stochastic simulator for four combinations of the diffusion coefficients and dissociation constants labeled in (A). (C) Number of cells killed depending on the nanoparticle formulation with k_a fixed and k_d varying. (D) Representative nanoparticle formulations identified in the literature. (E) Minimum injected dose of chemotherapy required to theoretically kill all cells in the simulated scenario for each nanoparticle formulation. Copyright ©2013 nanotoday.

Ω, a bounded convex open subset of $\mathbb{R}^2$ with Lipschitz continuous boundary $\partial\Omega$. Each agent switches probabilistically between an *active* state, during which it explores the domain with a combination of deterministic and random motion, and a *passive* state, during which it stops to take a measurement. The deterministic motion is governed by a time-dependent velocity $\boldsymbol{v}(t) \in \mathbb{R}^2$, and the random motion is represented as diffusion with an associated diffusion coefficient $D(\boldsymbol{x})$. The Diffusion can model a probabilistic search, exploration, and tracking strategy or stochasticity arising from sensor and actuator noise. An agent switches from the active state to the passive state at a time-dependent probability rate $k_1(t)$, and it switches back to the active state at a fixed probability rate k. In summary, each agent's position satisfies the following stochastic process:

$$\boldsymbol{dX}(t) = Y(t)\left(\boldsymbol{v}(t)dt + \sqrt{2D}\boldsymbol{dB}(t)\right) + d\phi(t), \tag{1.25}$$

where $Y(t)$ is another stochastic process that controls the transition between active and passive states, which satisfies:

$$\begin{aligned} P\left(Y(t+h) = 0 | Y(t) = 1\right) &= k_1(t)h + o(h^2) \\ P\left(Y(t+h) = 1 | Y(t) = 0\right) &= kh + o(h^2). \end{aligned}$$

The problem is stated as following.

Problem 1.3.1. *Denote $\mu_{\boldsymbol{X}(t)}$ be the distribution of* (1.25)*, and let $F(\boldsymbol{x})$ be a scalar field, which induces the target distribution $\mu_F = d\boldsymbol{x}\frac{F}{\int_\Omega F(\boldsymbol{y})\boldsymbol{dy}}$. How to design the control parameters $\boldsymbol{v}(t), D(\boldsymbol{x}), k_1(t), k$, such that $\mu_{\boldsymbol{X}(t)}$ converge weakly to μ_F, as $t \to \infty$?*

To this end, the authros develop the corresponding ADR system

$$\begin{cases} \dfrac{\partial y_1}{\partial t} = -\boldsymbol{v}(t)\nabla y_1 + \Delta(Dy_1) - k_1(t)y_1 + ky_2 \\ \dfrac{\partial y_2}{\partial t} = k_1(t)y_1 - ky_2 \end{cases} \tag{1.26}$$

with no-flux boundary condition

$$\boldsymbol{n} \cdot (\nabla(Dy_1) - \boldsymbol{v}y_1) = 0.$$

Theorem 1.3.2. *Denote* $y_1(\boldsymbol{x},t), y_2(\boldsymbol{x},t)$ *be the solution of the system* (1.26)*, and WLOG* $\int_\Omega y_1 \boldsymbol{dx} \equiv 1$*. Let* $\boldsymbol{v}(t) \equiv 0, k_1(t) \equiv 0, k = 0$*, and apply the following control law*

$$D(\boldsymbol{x}) = \frac{1}{F(\boldsymbol{x})}, \ x \in \Omega, \tag{1.27}$$

then $y_1(\boldsymbol{x},t)$ *converges in* L^1 *norm to* $F(\boldsymbol{x})$*, as* $t \to \infty$*.*

The proof of theorem 2.2.2 will be presented in more detail in chapter 2. According to theorem 2.2.2, the solution of the ADR system (1.26) will converge to target distribution with certain control law, hence in order to solve problem 1.3.1 we need $\mu_{\boldsymbol{X}(t)}$ converges weakly to $y_1(\boldsymbol{x},t)\boldsymbol{dx}$, which will be proved in chapter 3.

The field coverage control law is validated in two different simulated scenarios. Figure 1.4 and 1.5 show the target scalar fields, $F_1(\boldsymbol{x})$, and the dynamic distribution of swarm in both scenarios. The scalar fields $F_1(\boldsymbol{x})$ are numerically constructed as shown in the lower right of both figures.

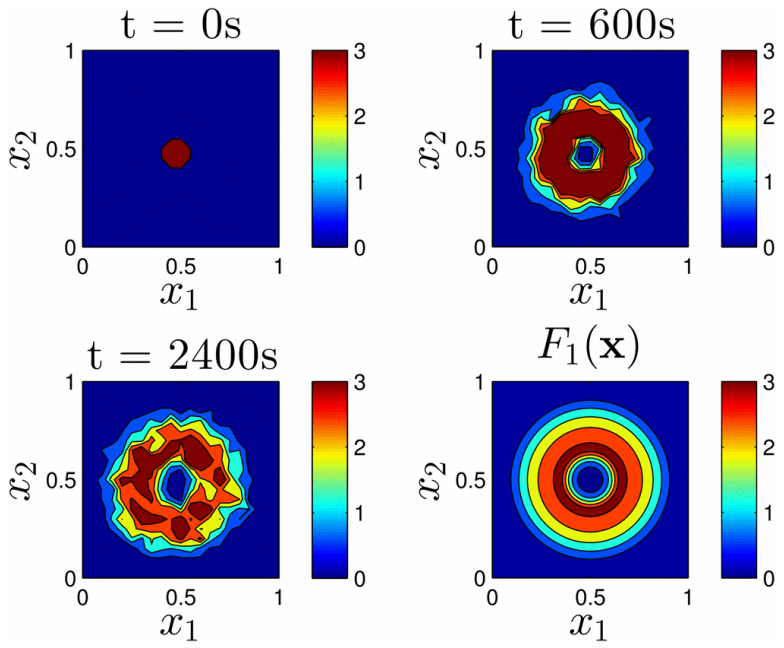


Figure 1.4: Simulated agent densities at three times t and the underlying scalar field with ring pattern. Copyright © 2016 IEEE.

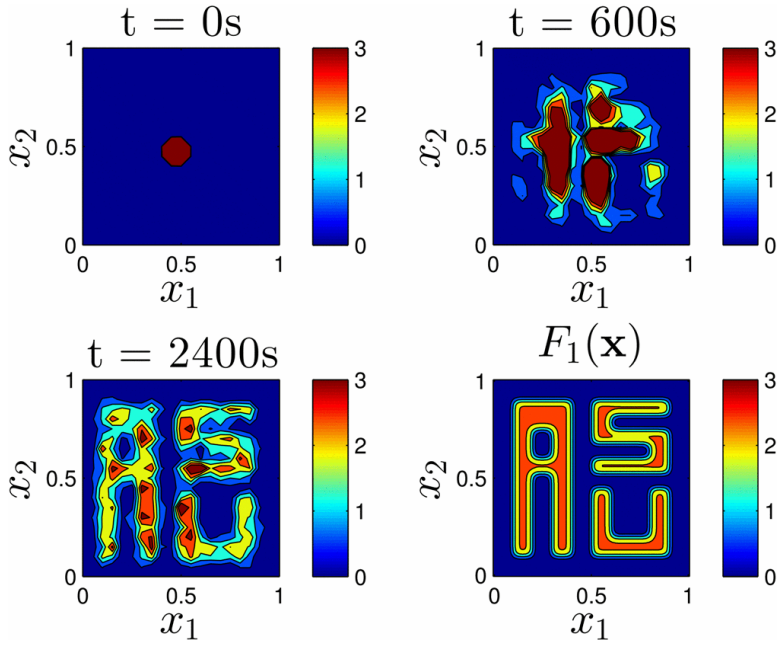


Figure 1.5: Simulated agent densities at three times t and the underlying scalar field with letter 'ASU'. Copyright © 2016 IEEE.

1.3.3 Robotic Swarm Pollination

In [EB15], K. Elamvazhuthi et al. studied an optimal control framework for deploying swarms of robot bees to pollinate a flower field. The goal is to achieve a target pollination distribution with effective control on resource-constrained robots that cannot localize or communicate with each other.

Consider deploying a swarm of N robots into the flower field Ω. Each member of the swarm performs the following actions during a flight. Upon deploying, each robot flies with a time-dependent velocity $\boldsymbol{v}(t)$, and a random movement with diffusion coefficient D, which arises from inherent noise due to sensor and actuator errors. The position $\boldsymbol{X}(t)$ at time t of each robot satisfies the following SDE, which is the microscopic model.

$$\boldsymbol{dX}(t) = \boldsymbol{v}(t)dt + \sqrt{2D}\boldsymbol{dB}(t). \tag{1.28}$$

The above SDE is simulated by

$$\boldsymbol{X}^i_{m+1} = \boldsymbol{X}^i_m + \boldsymbol{v}(t_m)\Delta t + \sqrt{2D\Delta t}\Delta \boldsymbol{Z}^i_m,\ 1 \leq i \leq N,$$

where X_m^i denotes the position of i^{th} robot at m^{th} time step , Z_m^i are independent, normally distributed random variables with zero mean and unit variance in $\mathbb{R}^2$.

Each robot has two states: F, flying state; H, hovering state. While a robot is flying over a row with flowers of type i $(i = 1, 2)$, it decides with a time-dependent probability per unit time $k_i(t)$ to pause at a flower in its sensing range and hover for pollination. The robot resumes flying with a fixed probability per unit time k_f , which determines the time taken to pollinate. The pollination is measured by another species: P, whose number increases by 1 whenever a robot transits from flying state to hovering state. The reaction network is:

$$\begin{aligned} F &\xrightarrow{k(t)} H + P \\ H &\xrightarrow{k_f} F. \end{aligned}$$

Denote the flower bed as $\Gamma = \Gamma_1 \cup \Gamma_2$, where Γ_i representing the region that flower of type i occupies, and denote the target distribution as $y_\Omega(\boldsymbol{x})$. A simple example of $y_\Omega(\boldsymbol{x})$ is

$$y_\Omega(\boldsymbol{x}) = 1 \text{ in } \Gamma \text{ and } 0 \text{ in } \Omega - \Gamma,$$

which requires uniform pollination over the flower bed.

The *optimal control problem* [EB15] aims to achieve $y_\Omega(\boldsymbol{x})$ by designing the control parameters $\{\boldsymbol{v}(t) = (v_1(t), v_2(t)), k_1(t), k_2(t)\}$. To this end, consider the following ADR system, or the macroscopic model:

$$\begin{cases} \dfrac{\partial F}{\partial t} = -\boldsymbol{v} \cdot \nabla F + D\Delta F - \displaystyle\sum_{i=1}^{2} k_i \mathbb{1}_{\Gamma_i} F + k_f H, \\ \dfrac{\partial H}{\partial t} = \displaystyle\sum_{i=1}^{2} k_i \mathbb{1}_{\Gamma_i} F - k_f H, \\ \dfrac{\partial P}{\partial t} = \displaystyle\sum_{i=1}^{2} k_i \mathbb{1}_{\Gamma_i} F, \end{cases} \tag{1.29}$$

with no-flux boundary condition

$$\boldsymbol{n} \cdot (D\nabla F - \boldsymbol{v}F) = 0.$$

The authors formulate the optimal control problem as following.

Problem 1.3.3 (Optimal Control Problem)**.** *Let $u = (v_1(t), v_2(t), k_1(t), k_2(t))$ be the collection of all control parameters, then the optimization problem can be framed as*

$$\min_{u \in U_{ad}} J(u) \;=\; \frac{1}{2}\|P(\cdot, T) - y_\Omega\|^2_{L^2(\mathbb{R}^2)} + \frac{\lambda}{2}\|u\|_{L^2(0,T)},$$

under the constraint of system (1.29)*. Here $U_{ad} = \{u \in L^2(0,T)^3 : u_i^{min} \leq u_i(t) \leq u_i^{max} \;\; \forall t \in (0,T)\}$ is the set of admissible control inputs.*

In [EB15], problem 1.3.3 is solved in a gradient decent manner, which requires the following proposition.

Proposition 1.3.4. *Denote the map $M : u \to \boldsymbol{y} = (F, H, P)$, introduced by system* (1.29)*. Then the map M is Gateaux differentiable at every $u \in U_{ad}$, i.e. its Gateaux derivative evaluated at $h \in U_{ad}$*

$$\lim_{\epsilon \to 0} \frac{M(u + \epsilon h) - M(u)}{\epsilon}$$

exists.

The proof of proposition 1.3.4 can be found in [EB15].

In the following simulation, a swarm of robots is tasked to achieve a specified spatial distribution of flower visits over five crop rows, with flower of type 1 occupies the 1st, 3rd, and 5th row, and flower of type 2 occupies 2nd and 4th row. In [EB15], two objectives are considered: one in which visits were required along all five crop rows (Objective 1), and another in which they were required only along two of the crop rows (Objective 2). For both objectives, the authors simulate an environment with and without obstacles. Hence there are 4 scenarios in total. For each scenario, $N = 1000$ robots are deployed into a normalized domain of size 1 m $\times$ 1 m. The transition rate k_f is set to be $0.2s^{-1}$, so that the expected pollination time is 5 s, which is a realistic value for certain bee species [BNH11]. In the optimization, the robot speed is bounded in $[0, 10]$ $m \cdot s^{-1}$, and the transaction rates $k_i(t)$ are bounded in $[0, 1.25]$ s^{-1}. In the microscopic model, the field is equally discretized into 21×21 cells, and the author compute number of pollinations in each cell, where each robot is considered as a point mass. In the macroscopic model, the ADR system is numerically solved

over a grid 51×51 cells. Figure 1.6 shows the decay of the objective functions of the optimal control scheme in the 4 scenarios over time. Figure 1.7 and 1.8 show the distribution of flower visits of microscopic and macroscopic models for objective 1, without and with obstacles, respectively. Figure 1.9 and 1.10 show the distribution of flower visits of microscopic and macroscopic models for objective 2, without and with obstacles, respectively.

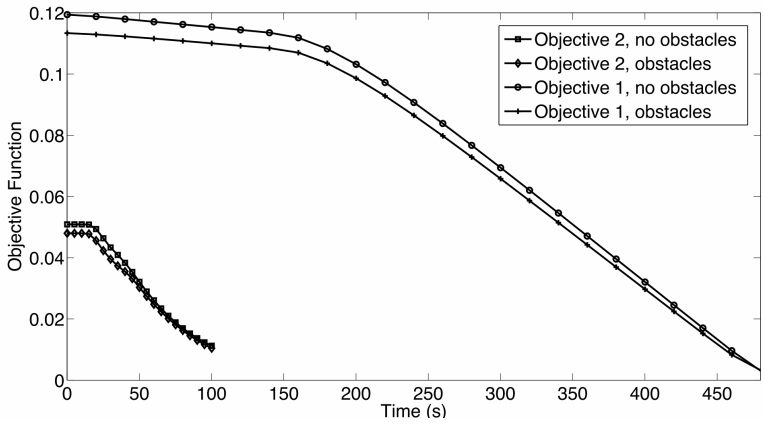


Figure 1.6: Objective function over time for objective 1 and 2, with or without obstacles. Copyright © 2015 IEEE.

1.4 Discussion

In section 1.3, we presented numerical simulations of a few tasks, including scalar field estimation, and robotic swarm pollination. However, there are many real-world complexities that are not accounted in the model, some of which are ignored in order to simplify the model. For example, due to the finite clock rate of the microprocessor, the robots have a finite maximum speed, while in the simulation they do not; The friction between robots' wheel and body makes the robots 'inertial', which hinders the accuracy of the robots' motion at low speed; The robots' path may deviate from the desired one due to the sensor error. All of these uncertainties makes us doubt whether our control law or framework is robust in the real world case. In chapter 2, we present test bed experiments with a real robot to study the control's effectiveness and robustness.

In this chapter we reviewed the proof of the Kolmogorov theorem, and also presented a simulation using a point mass approach to verify the theorem. We viewed each particle as a point mass and computed the density by counting the number of point masses in a region. This method is sub-optimal from two aspects. First, the accuracy is very sensitive to the choice of the region. If the region is too small, it may not contain any particle; if the region is the whole space, the error will always be zero. Second, it does not provide a global measurement between the microscopic and macroscopic models. In chapter 3, we solve these problems by introducing a 'blob' approach, in which we treat each particle as a blob with finite volume. The new approach generates an *actual density function* for the microscopic model, which then leads to a global error metric as the L^1 norm of the difference between the densities from the microscopic and macroscopic models.

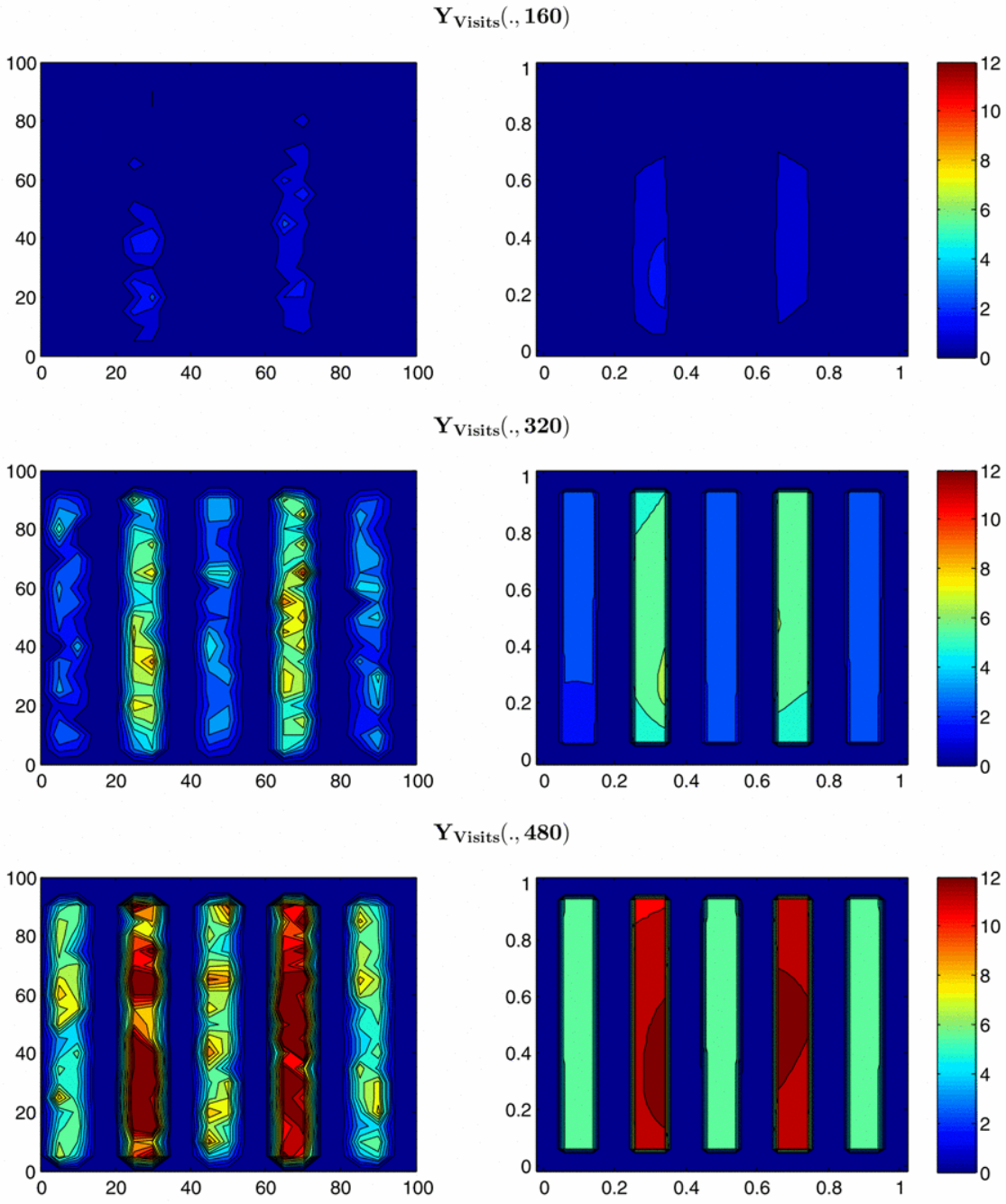


Figure 1.7: Distribution of flower visits at three times for objective 1, no obstacle. Left: microscopic model; Right: macroscopic model. Copyright © 2015 IEEE.

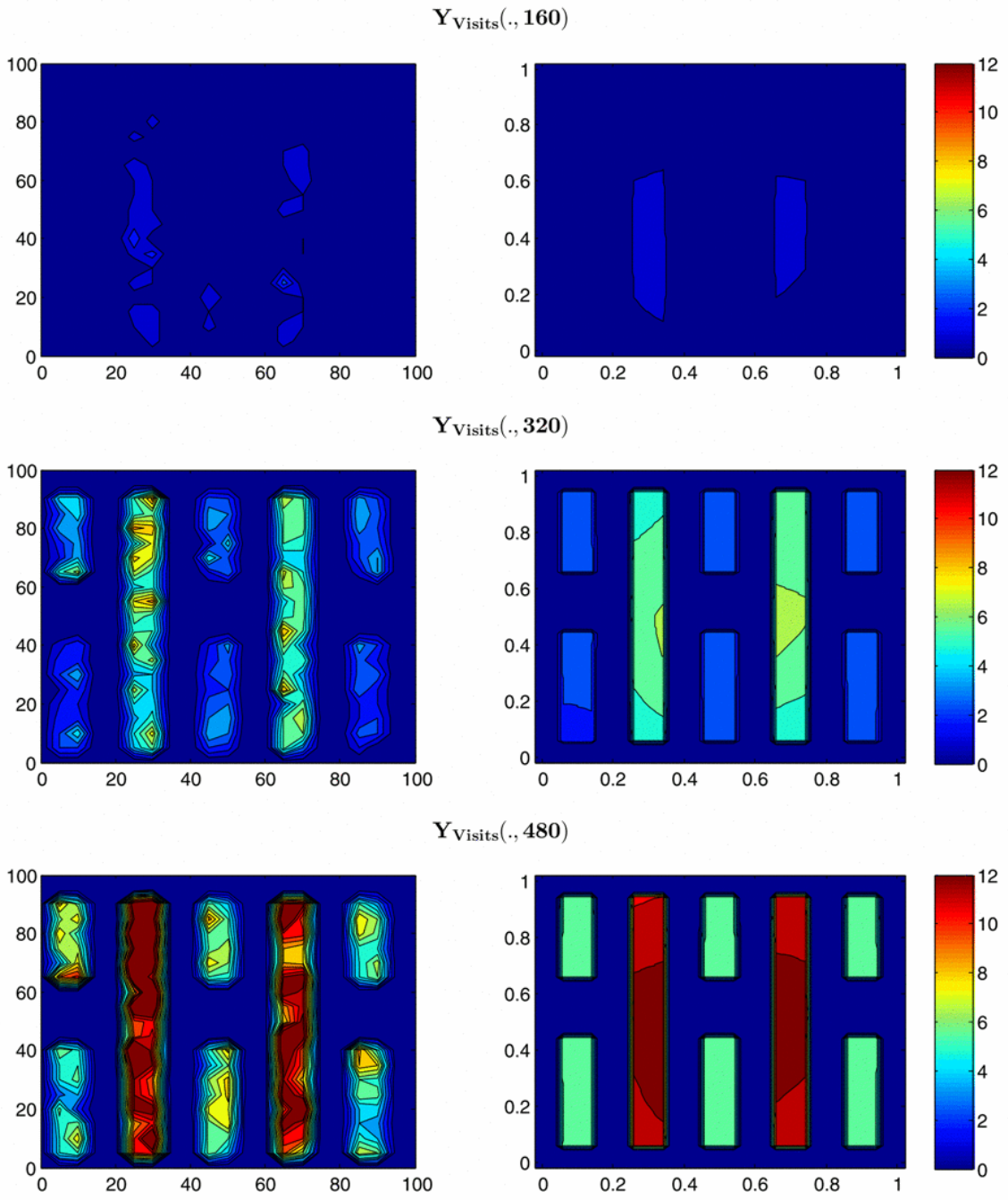


Figure 1.8: Distribution of flower visits at three times for objective 1, with obstacles. Left: microscopic model; Right: macroscopic model. Copyright © 2015 IEEE.

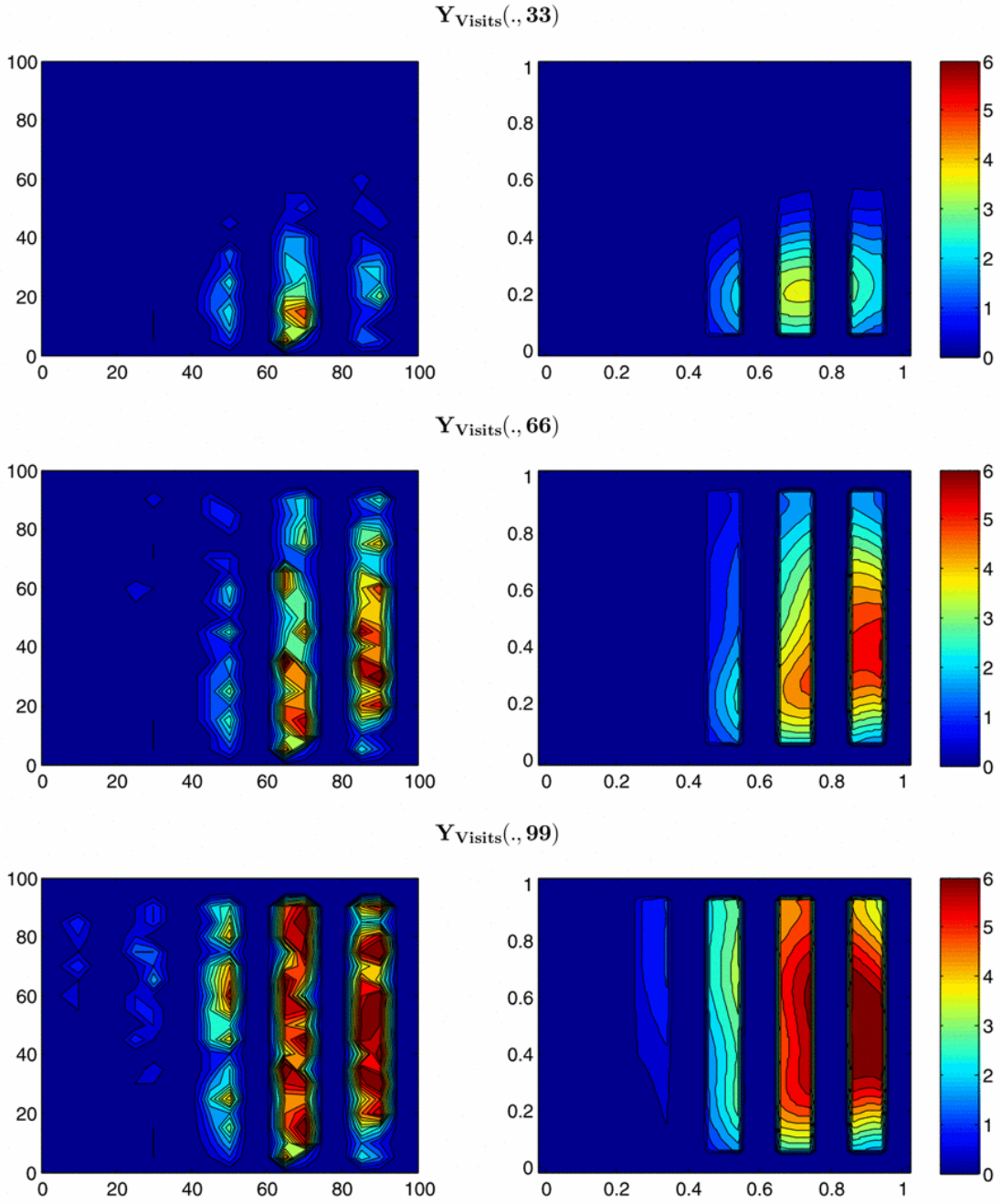


Figure 1.9: Distribution of flower visits at three times for objective 2, no obstacle. Left: microscopic model; Right: macroscopic model. Copyright © 2015 IEEE.

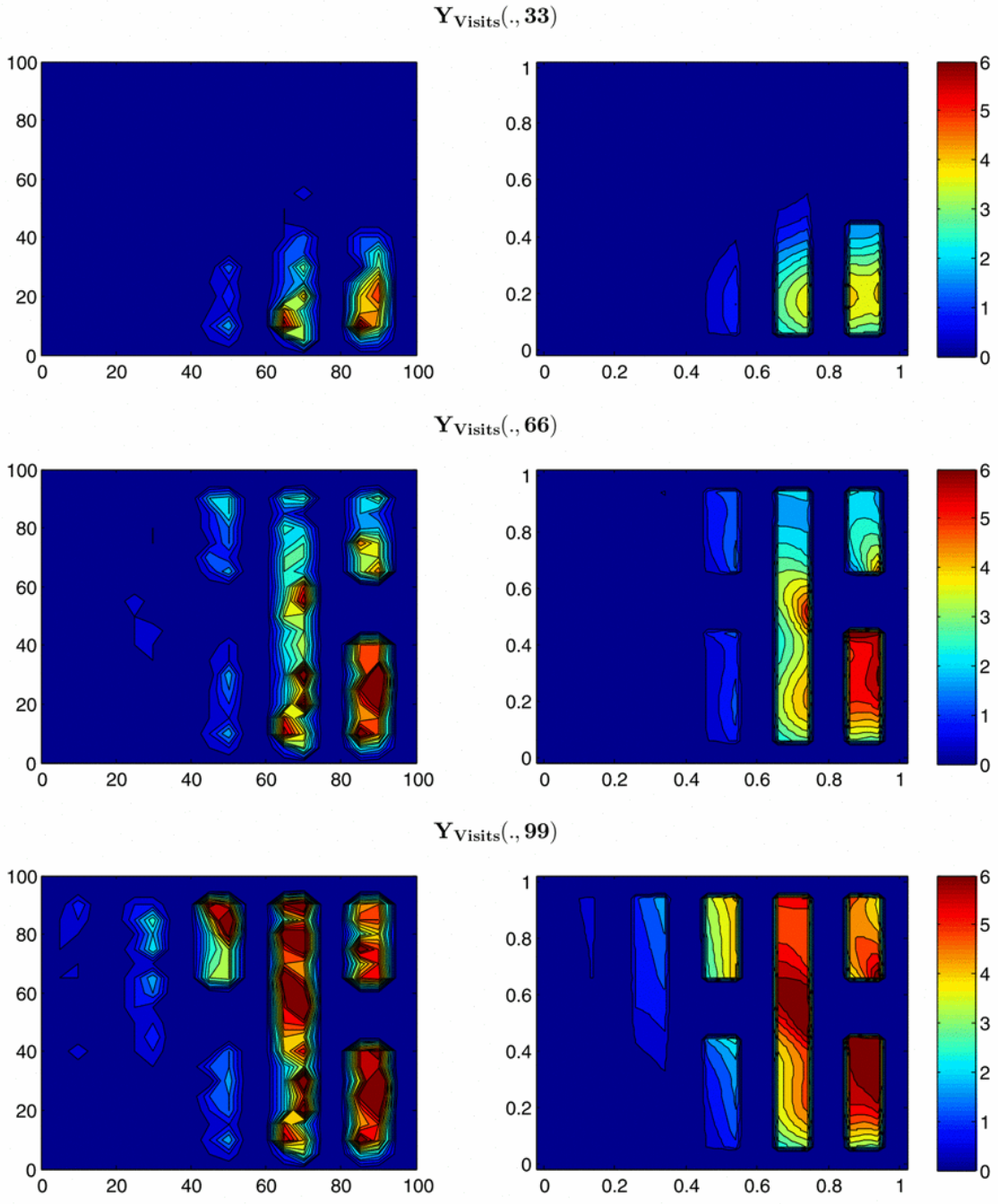


Figure 1.10: Distribution of flower visits at three times for objective 2, with obstacles. Left: microscopic model; Right: macroscopic model. Copyright © 2015 IEEE.

CHAPTER 2

Decentralized Stochastic Control of Robotic Swarm

2.1 Introduction

Distributed control laws for multi-agent coverage strategies have been widely investigated [CMK04], [SRS09]. Applications of coverage strategies include environmental monitoring, surveillance policies, source localization problems [HMG02], and vehicle scheduling [PFB11]. In this chapter, we explore a stochastic approach in controlling the distribution of a robotic swarm toward a predefined target distribution in a bounded domain.

In many modern tasks, individual robots may often lack capabilities that are assumed to be essential for robot control, which is a trade-off between the cost of the robotic swarm and the accuracy of the control. For example, In [BKN11], S. Berman et al. proposed using robotic bees to perform commercial pollination, who can only acquire local information. In [MCF13], K. Y. Ma et al. introduces a set of insect-sized flying robots, among which even the most advanced robot is extremely weight-constrained. For this reason, we are interested in a resource-constrained robotic swarm without localization or communication capabilities. In our model, the robots only have the ability to measure a scalar field from the environment, from which they follow a control law that drives them toward a target distribution.

The use of stochastic methods in a similar vein has seen much work in literature recently [AB12], [APB14], [BCH13]. An important characteristic of many of these methods has been the index-free/permutation-invariant nature of the control laws, which can be beneficial for scalability in design [BK04], [KE11]. The reason for using stochastic control is two-fold. On the one hand, the control of robots is not absolutely accurate due to inherent sensor and actuator errors, which can be modeled as stochastic motion; on the other hand, stochastic

control can be more stable and less sensitive to errors [DBK12] .

This chapter is organized as following. Section 2.2 presents a brief mathematical background of the control law. Secion 2.3 shows the detail of our computational simulation in MATLAB. Section 2.4 describes our experiments on the testbed of our laboratory.

2.2 Control Law and Analysis

2.2.1 Problem formulation

In [EAB16], Karthik et al. studied a swarm of N robots that are deployed into a domain $\Omega \subset \mathbb{R}^2$. Each robot explores the domain with random motion, which is modeled as a 2D Brownian motion with a location-dependent diffusion coefficient $D(\boldsymbol{x})$. The robots' position is denoted by $\boldsymbol{X}^i(\cdot) \in \Omega(1 \leq i \leq N)$, which is a group of stochastic processes with independent and identically distribution (i.i.d) that satisfies the following SDE:

$$\boldsymbol{dX}(t) = \sqrt{2D(\boldsymbol{X}(t))}\boldsymbol{dB}(t) + d\phi(t), \tag{2.1}$$

where $d\phi(t)$ represents the robots' mirror reflection at the boundary $\partial\Omega$, and 0 elsewhere.

We now present the problem of coverage of a predefined scalar field $F(\boldsymbol{x}) : \Omega \to \mathbb{R}^+$. We obtain the target distribution μ_F by normalizing F:

$$\mu_F = d\boldsymbol{x}\frac{F}{\int_\Omega F(\boldsymbol{y})\boldsymbol{dy}}.$$

We desire to find a diffusion coefficient $D(\boldsymbol{x})$, so that the distribution of SDE (2.1), which we denote as $\mu_{X(t)}$, converges weakly to μ_F, as $t \to \infty$?

Since (2.1) governs the micro level of the swarm, i.e. the behavior of each individual robot, we call (2.1) *the microscopic model*. According to definition 1.2.1, the Fokker-Planck equation of (2.1) is

$$\frac{\partial y}{\partial t} = \Delta(Dy) \tag{2.2}$$

with the no-flux boundary condition $\boldsymbol{n}\cdot\nabla(Dy) = 0$, which corresponds to the mirror reflection $d\phi(t)$. Since (2.2) governs the group behavior of the swarm, we call (2.2) *the macroscopic model*.

Now according to corollary 1.2.4, we know that for any region $\Sigma \subset \Omega$, we have

$$E\left(\mathbb{1}_\Sigma(\boldsymbol{X}(t)\right) = \frac{1}{N}\sum_i E\left(\mathbb{1}_\Sigma(\boldsymbol{X}_t^i)\right) = \int_\Sigma y(\boldsymbol{x},t)\boldsymbol{dx},$$

where we assumed total mass to be unit: $\int_\Omega y(\boldsymbol{x},t)\boldsymbol{dx} = 1$. Hence $\mu_{X(t)} = y(\boldsymbol{x},t)\boldsymbol{dx}$ for all $t > 0$, i.e. the macroscopic model and the microscopic model agree with each other. So the above problem can be reduced to the following problem.

Problem 2.2.1. *Determine whether there exists a control law $D(\boldsymbol{x})$, such that the solution to the macroscopic model* (2.2) $\lim_{t\to\infty} y(\boldsymbol{x},t) = c \cdot F(\boldsymbol{x})$ *for some constant c.*

Toward this end, our specific choice of control law is given by

$$D(\boldsymbol{x}) = \frac{1}{F(\boldsymbol{x})}. \tag{2.3}$$

In the rest of this section, we solve problem 2.2.1 by proving the following theorem.

Theorem 2.2.2. [EAB16] *With the choice of the control law* (2.3)*, we have*

$$\int_\Omega |y(\boldsymbol{x},t) - c\cdot F(\boldsymbol{x})|\boldsymbol{dx} \le Me^{-\lambda t},$$

for some constant $c > 0, M > 1, \lambda > 0$, and all $t > 0$.

2.2.2 Preliminaries

In this subsection we recall some standard notions from theory of operator semigroups. Let H be a Hilbert space and $L(H)$ be the space of bounded operators in H.

Definition 2.2.3. *A family of operators $(\mathbf{T}(t))_{t\ge 0}$ in $L(H)$ is a strongly continuous semigroup on H if*

- $\mathbf{T}(t) = \mathbf{I}$,
- $\mathbf{T}(t+\tau) = \mathbf{T}(t)\mathbf{T}(\tau)$,
- $\lim_{t\to\infty} \mathbf{T}(t)z = z,\ \forall z \in H$.

Definition 2.2.4. *The linear operator $A : \mathcal{D}(A) \to H$, defined by*

$$\mathcal{D}(A) = \left\{ z \in H : \lim_{t \to 0} \frac{\mathbf{T}(t)z - z}{t} \text{ exists } \right\},$$

$$Az = \lim_{t \to 0} \frac{\mathbf{T}(t)z - z}{t}, \ \forall z \in \mathcal{D}(A)$$

is called infinitesimal generator of semigroup $\mathbf{T}(t)$.

Definition 2.2.5. *An unbounded linear operator $A : \mathcal{D}(A) \to H$ is said to be dissipative if*

$$Re\langle Az, z\rangle \leq 0, \ \forall \mathcal{D}(A). \tag{2.4}$$

We define the space of square integrable real-valued measurable functions with respect to function $w(\boldsymbol{x})$ on, $L^2_w(\Omega)$, with the weighted 2-norm as

$$L^2_w(\Omega) := \left\{ f(\boldsymbol{x}) | \|f\|_{2w} = \left(\int_\Omega |f(\boldsymbol{x})|^2 w(\boldsymbol{x}) \boldsymbol{dx} \right)^{1/2} < \infty \right\},$$

and the induced inner product $(\ \cdot \ , \ \cdot \ )_w : L^2_w(\Omega) \times L^2_w(\Omega) \to \mathbb{R}$ as

$$(f, g)_w := \int_\Omega f(\boldsymbol{x}) g(\boldsymbol{x}) w(\boldsymbol{x}) \boldsymbol{dx}.$$

Define $H^2_w(\Omega)$ as

$$H^2_w(\Omega) := \left\{ f \in L^2_w(\Omega) : \frac{\partial^2 f}{\partial x^2_\alpha} \in L^2_w(\Omega), \forall \alpha \in \{1, 2\}. \right\},$$

which is equipped with the inner product

$$(f, g)_{H^2_w} = \int_\Omega f(\boldsymbol{x}) g(\boldsymbol{x}) w(\boldsymbol{x}) \boldsymbol{dx} + \sum_{j=1}^{2} \sum_{i=1}^{2} \int_\Omega \frac{\partial^j (wf)}{\partial x^j_i}(\boldsymbol{x}) \frac{\partial^j (wg)}{\partial x^j_i}(\boldsymbol{x}) \boldsymbol{dx}.$$

Then it is easy to see that $H^2_w(\Omega)$ is a Hilbert space. Note that for $w \equiv 1$, $H^2_w(\Omega)$ is the same as the usual Sobolev space $H^2(\Omega)$. We can then write the macroscopic model in the form of operator $A : \mathcal{D}(A) \to H^2_w(\Omega)$ as following.

$$\begin{aligned} &\frac{\partial y}{\partial t} = Ay \\ &y(\boldsymbol{x}, 0) = y_0(\boldsymbol{x}) \\ &Af = \Delta(wf), \ \forall f \in \mathcal{D}(A), \end{aligned} \tag{2.5}$$

where $\mathcal{D}(A) = \{ f \in H^2_w(\Omega) : \boldsymbol{n} \cdot \nabla(wf(\boldsymbol{x})) = 0, \ \forall \boldsymbol{x} \in \partial\Omega \}$.

2.2.3 Analysis of Control Law

Proposition 2.2.6. *Defined in* (2.5)*, A is a dissipative operator on $L^2_w(\Omega)$ and generates a strongly continuous semigroup on $L^2_w(\Omega)$.*

Proof. Using integration by parts and the boundary condition, $(Az, z)_w = -\int_\Omega \|\nabla(wf)\|^2 \boldsymbol{dx} \leq 0$, so A is dissipative by definition 2.2.5. Next we define a bilinear form

$$B(u, v) = ((I - A)u, v)_{H^1_w}. \tag{2.6}$$

Then we have

$$|B(u, v)| \leq c\|u\|_{H^1_w(\Omega)}\|v\|_{H^2_w(\Omega)} \quad \text{(continuity)},$$
$$|B(u, u)| \geq c'\|u\|^2_{H^2_w(\Omega)} \quad \text{(coerciveness)}.$$

By Lax-Milgram theorem, for each $f \in L^2_w(\Omega)$ there exists a unique solution $u \in H^1_w(\Omega)$ such that

$$B(u, v) = (f, v)_{H^1_w}$$

for all $v \in H^1_w(\Omega)$. This means that the range of operator $(I - A)$ is the whole space $L^2_w(\Omega)$. Therefore, the result follows from the dissipativeness of A and Corollary 3.2 in Chapter II of [EN99]. □

Suppose the semigroup that A generates is $\mathbf{T}(t)$, then theorem 2.2.2 can be restated as following.

Theorem 2.2.7. *With the choice of the control law* (2.3)*, we have*

$$\int_\Omega |\mathbf{T}(t)y_0 - c/w| \boldsymbol{dx} \leq Me^{-\lambda t},$$

for some constant $c > 0, M > 1, \lambda > 0$, and all $t > 0$.

Proof. Define a linear map $R_t : \mathcal{D}(A) \to \mathbb{R}$ by

$$R_t(u) = \int_\Omega (\mathbf{T}(t)u - u)\, \boldsymbol{dx}.$$

Then, by conservation of mass, it is easy to deduce that $R_t(u) \equiv 0$ for all $u \in \mathcal{D}(A), t > 0$. Due to the boundedness of R_t, and that $\mathcal{D}(A)$ is dense in $L^2_w(\Omega)$, R_t can be extended to a bounded linear operator from $L^2_w(\Omega)$ to $\mathbb{R}$. Since the Laplace operator Δ has eigenvector, constant function, for the eigenvalue 0, A has a unique one-dimensional subspace of eigenvectors, spanned by the function $1/w = 1/D(\boldsymbol{x})$, corresponding to the eigenvalue 0, and 0 is the first-order pole of A. The equilibrium of solution $y(\boldsymbol{x}, t)$ is then the eigenvector $1/D(\boldsymbol{x})$ as $t \to \infty$, according to [EN99], Chapter V, Corollary 3.3. □

2.3 Simulation of Two Target Distribution Examples

In [LFE17], we simulate the microscopic model in MATLAB for two target distributions that are chosen as performance benchmarks for our testbed experiments, presented in Section 2.4. We specifically choose two scalar fields of different patterns, ring and row, in our simulation. The scalar fields are plotted in figure 2.1. The equation for the scalar field is

$$F(\mathbf{x}) = \begin{cases} 36 & \text{if } \mathbf{x} \in \Omega \cap \Gamma, \\ 1 & \text{if } \mathbf{x} \in \Omega \setminus \Gamma, \end{cases}$$

where:

- $\Omega = \{\mathbf{x} : x_1 \in [0, w], x_2 \in [0, h]\}$,
- (ring) $\Gamma = \{\mathbf{x} : r_1^2 < (x_1 - \frac{w}{2})^2 + (x_2 - \frac{h}{2})^2 < r_2^2\}$,
- (rows) $\Gamma = \{\mathbf{x} : x_1 \in [d, 2d] \cup [3d, 4d] \cup [5d, 6d]\}$.

with (x_1, x_2) Cartesian coordinates of $\mathbf{x}$, $w = 48$in and $h = 70$in.

For T big enough, we discretize the time span of swarm deployment $[0, T]$ into M equal time steps.

$$0 = t_0 < t_1 < ... < t_M = T, \quad t_m = m\Delta t,$$

and simulate the microscopic model (2.1) by a difference equation

$$\boldsymbol{X}_i^{m+1} = \boldsymbol{X}_i^m + \sqrt{2D(\boldsymbol{X}_i^m)\Delta t}\boldsymbol{Z}_i^m, \tag{2.7}$$

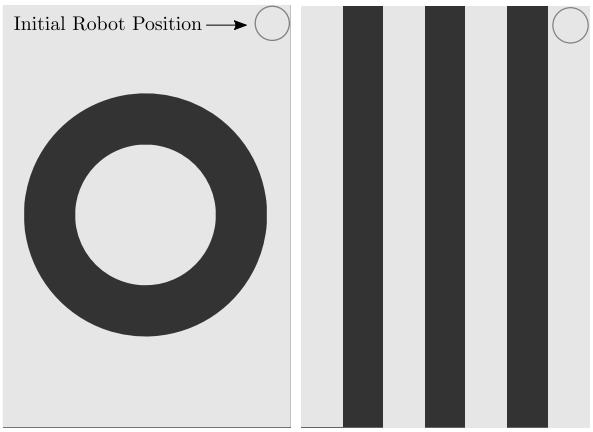


Figure 2.1: **The Scalar Fields.** Each pattern above specifies a scalar field, which induces a target distribution. Left: ring pattern. Right: row pattern. Copyright ©2017 IEEE.

where $\boldsymbol{X}_i^m$ denotes the position of i^{th} robot at m^{th} time step, and $\boldsymbol{Z}_i^m,\ \forall i, m$ is a group of independent, normally distributed random variables with zero mean and unit variance. A real robot has a finite maximum speed $v_{\max}$, hence the speed of the i^{th} robot at the m^{th} time step is defined as

$$v_i^m = \min\left(\sqrt{2D(\boldsymbol{X}_i^m)}\Delta t^{-\frac{1}{2}}|\boldsymbol{Z}_i^m|\ ,\ v_{\max}\right).$$

Moreover, a real robot with only a local scalar sensor cannot easily determine the orientation of the boundary relative to its current velocity, and thus cannot reflect specularly. Instead, when the robot senses contact with the boundary, it simply reverses direction for the duration of the time step. Therefore, if $\boldsymbol{X}_i^m$ calculated according to equation (2.7) is found to be outside the boundary, we calculate a coefficient $\alpha = \frac{\Delta t_a - \Delta t_b}{\Delta t_a + \Delta t_b}$, where Δt_a is the duration until the robot would reach the boundary at its present speed, and $\Delta t_b = \Delta t - \Delta t_a$ is the remainder of the time step. Then

$$v_i^m = \alpha \min\left(\sqrt{2D(\boldsymbol{X}_i^m)}\Delta t^{-\frac{1}{2}}|\boldsymbol{Z}_i^m|\ ,\ v_{\max}\right).$$

For both scalar fields, we choose $v_{\max} = 17.5$in/s to reflect the maximum speed of the physical robot, as described in Section 2.4, and $\Delta t = 0.5$s. Because the maximum achievable speed of the physical robot depends on the direction of movement, $v_{\max}$ is really the minimum, over all movement directions, of the maximum achievable speed. The time step $\Delta t = 0.5$s is chosen to be relatively small to avoid error due to time discretization, but large enough to

permit the robot to accelerate to each new velocity well within the time step.

Suppose $\rho(\boldsymbol{x}, t)$ is the density function of $\boldsymbol{X}(t)$ defined in SDE (2.1), which satisfies

$$\int_{\Sigma} \rho(\boldsymbol{x}, t) \boldsymbol{dx} = \int_{\Sigma} d\mu_{\boldsymbol{X}(t)}.$$

In Chapter 1 we compare the actual distribution of the swarm at a specific time T with $\rho(\boldsymbol{x}, T)$ by counting the number of robots in a specific region Σ. This comparison is sub-optimal from two aspects. First, this method is restricted in one region, while a good metric should be able to measure the difference between these two distributions. Second, this comparison is sensitive to the choice of Σ. For example, if we choose Σ to be the whole region Ω, then the difference will be 0 under any control law. In order to have a better comparison, we define the *actual density field* by

$$y_{\delta}(\boldsymbol{x}, t) = \frac{1}{N} \sum_{i=1}^{N} G_{\delta}(\boldsymbol{x} - \boldsymbol{X}^i(t)), \tag{2.8}$$

where δ represents the radius of the region in which the robot effectively performs its task, and G_{δ} is the Gaussian function

$$G_{\delta}(\boldsymbol{x}) = \frac{1}{2\pi\delta^2} \exp\left(-\frac{|\boldsymbol{x}|^2}{2\delta^2}\right).$$

That is, we center a Gaussian 'blob' at each robot's position, instead of viewing them as a swarm of point mass.

2.3.1 Simulation Results

We show the simulation results of $y_{\delta}(\boldsymbol{x}, T)$, for large enough T. We choose $N = 200, \delta = 2\text{in}$ and $T = 800s$ for both target distributions of figure 2.1. Figure 2.2 shows the positions of robots and the actual density field $y_{\delta}(\boldsymbol{x}, T)$ at different time. To evaluate the result, we define the error

$$e_{\delta}(t) = \int_{\Omega} |y_{\delta}(\boldsymbol{x}, t) - F(\boldsymbol{x})| \boldsymbol{dx}.$$

Since $N = 200$ is not a large number, we first compute an upper bound of the minimum error that one can achieve in both scenarios by manually distributing the robots in our domain.

Since it is mathematically impractical to rigorously find out the best position of each robot, we choose the most 'reasonable' configuration according to each scalar field. This minimum error would be seen as the best performance. The two simulations yield $e_\delta(T) = 0.4295$, with minimum error value 0.3089 for the ring pattern, and $e_\delta(T) = 0.5366$, with minimum error value 0.3549 for the row pattern.

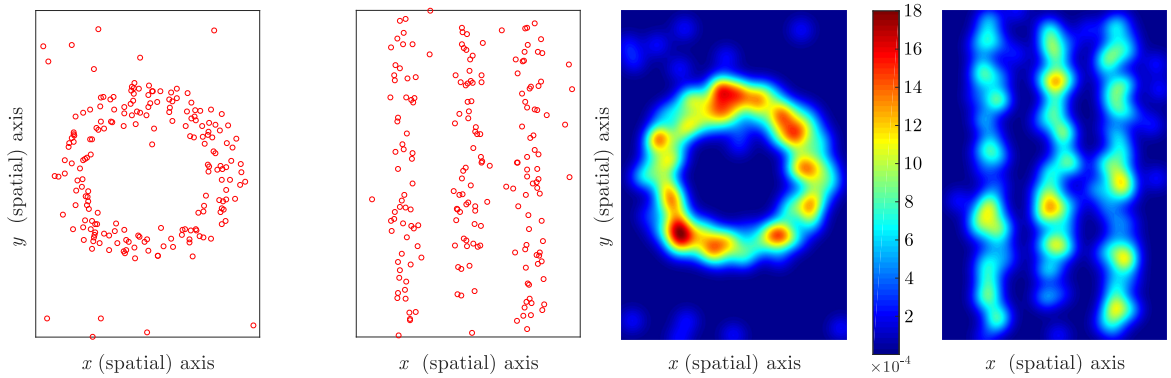


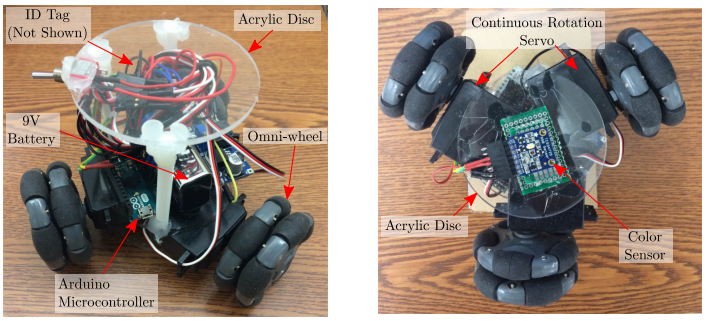
Figure 2.2: Top: Robot positions at $t = 601.5$s for ring distribution with $N = 200$ robots (left) and at $t = 361.5$s for row distribution with $N = 200$ robots (right).
Bot: Gaussian blob function $y_{\delta=2\text{in}}(t = 601.5\text{s})$ for ring distribution (left) and $y_{\delta=2\text{in}}(t = 361.5\text{s})$ for row distribution (right). Copyright ©2017 IEEE.

2.4 Testbed Experiment

In [LFE17], the authors further test the robustness of our control law with a real robot in the lab.

2.4.1 Setting

We design and build a new holonomic drive robot, who can translate in any direction independent of its orientation [RF06], for our testbed experiment. The holonomic drive is realized by three 58mm Nexus Robot omni-wheel (RB-Nex-57), which rolls like a typical wheel but also slides freely relative to the floor along the axis of rotation. The complete robot is shown in figure 2.3.



(a) View from above (b) View from bottom

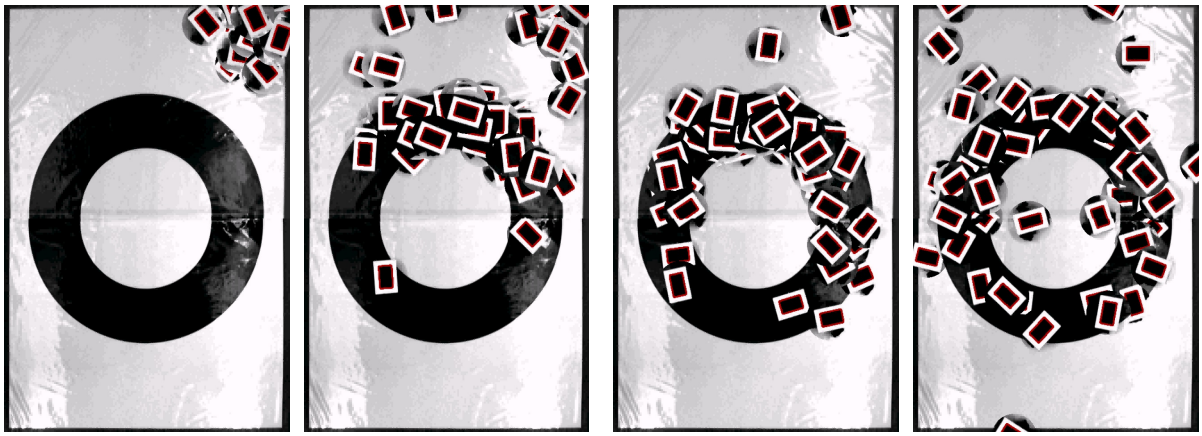
Figure 2.3: **The Kiwi-drive Robot.** For clarity of other components, identification tag visible in Figure 2.4 is not shown. Copyright ©2017 IEEE.

We achieve the behavior a swarm of many robots by performing many trials on one robot, which is valid on our spatial coverage experiment because robots have no communication abilities and thus are independent with each other. We also do not consider collision, due to the assumption that the domain Ω is large and our robot is very small compared to it. The scalar fields of figure 2.1 are printed in RGB color, which are then placed on the ground. The scalar field are sensed by a digital RGB color sensor (TCS34725) under the chassis of the robot: black represents a high value of the scalar field, white represents low value, and red represents the boundary. Two overhead cameras are used to collect data throughout time, which processes OpenCV in python to track the position of the centroid of the robot. The detection error between the position measured by the computer and the real position of the robot is within 2cm.

2.4.2 Results

Figure 2.4 shows the convergence of the 'swarm' to the desired distribution through time-lapse images in the scalar field of ring pattern. The results are obtained by running 1 robot 200 times rather than 200 robots at once. This avoids worrying about collision control. Each subfigure is produced by superposing the images of a random subset of 50 runs.

In order to evaluate the result, we take the centroid of the robots as their positions in each frame, so that each robot becomes a point. We then measure the actual density distribution



(a) $t \approx 0$ s (b) $t \approx 25$ s (c) $t \approx 80$ s (d) $t \approx 170$ s

Figure 2.4: The convergence of the swarm to target distribution over time. Notice that the robot is the same as Figure 2.3, and it is additionally obscured by identification tag, visible as a solid black rectangle within a white rectangle. Copyright ©2017 IEEE.

of our testbed experiment by Gaussian blobs, as in (2.8). The point mass distributions and the actual density distributions are shown in figure 2.5. The times chosen for the figure are those at which the error metric e_δ is minimized. The result verifies that the control law guides the robots toward the target distribution. However, the experiments did not run long enough to achieve a steady state, at which e^δ reaches an equilibrium. The concentration in the top right remains higher than elsewhere in the row pattern, since the top right corner is where all the robots started. Figure 2.6 plots the error metric over time $e_\delta(t)$ from both simulation and testbed experiment in the ring pattern. Despite many real-world complexities, the testbed experiment agrees with the simulation in the scale of the error metric at convergence, and even in the time of convergence. This justifies that our control law in robust in real world applications.

2.4.3 Extension: N convergence

In chapter 1, we observe the $N^{-1/2}$ convergence of the number of robots in a specific region. We show here that the error metric $e_\delta(t)$ has the same convergence rate with respect to N. In the ring distribution, we specifically choose five values of N: 10, 20, 40, 80, and 190,

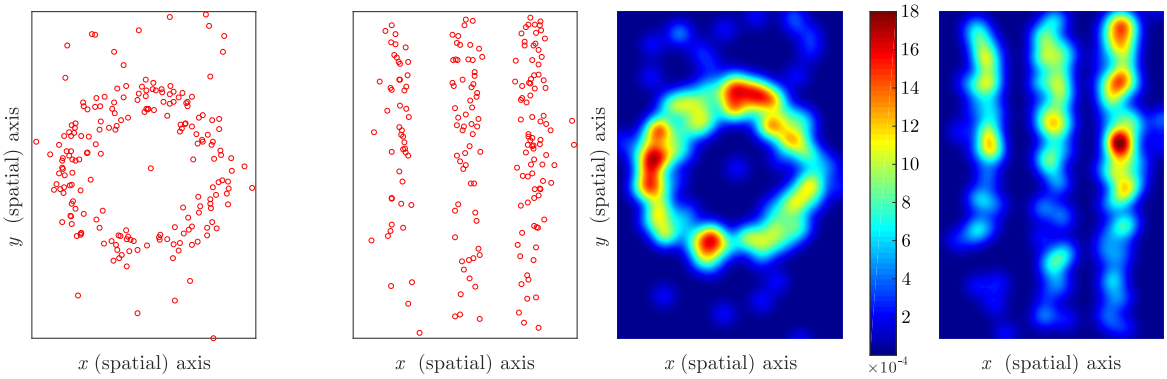


Figure 2.5: Robot positions at $t = 175.5$s for ring distribution (left) and at $t = 160.5$s for row distribution (right). Notably, there are two robots in the row pattern that does not show in the figure, because they are still outside the boundary and in the process of bouncing back in. Copyright ©2017 IEEE.

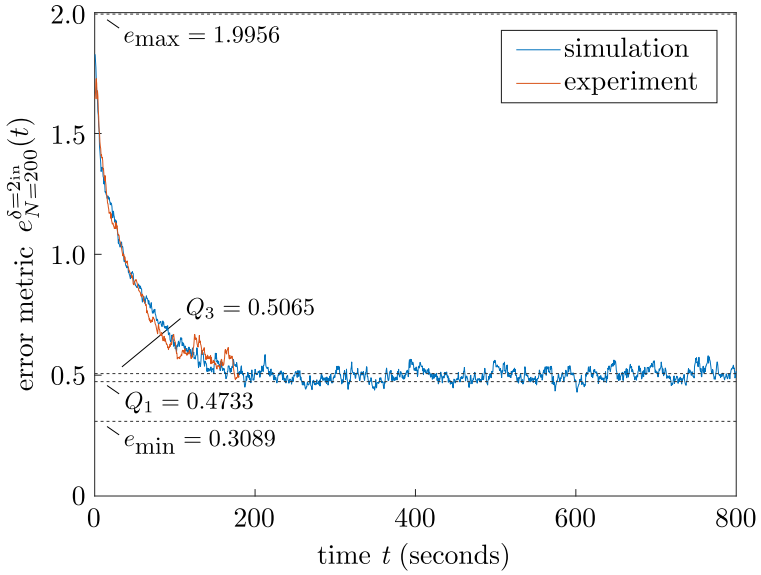


Figure 2.6: Error metric $e_\delta(t)$ over time for both simulation and physical experiment with the target distribution as the ring pattern. We also include Q_1 and Q_3, which are lines for the minimum and maximum error achieved by manually displacement, labeled $e_{\min}$ and $e_{\max}$, respectively. Copyright ©2017 IEEE.

and use the average of $e_\delta(t)$ of last 60 seconds. We plot the results of the simulation and the testbed experiment with respect to $N^{-1/2}$ in figure 2.7, and draw the best fit line of the simulation.

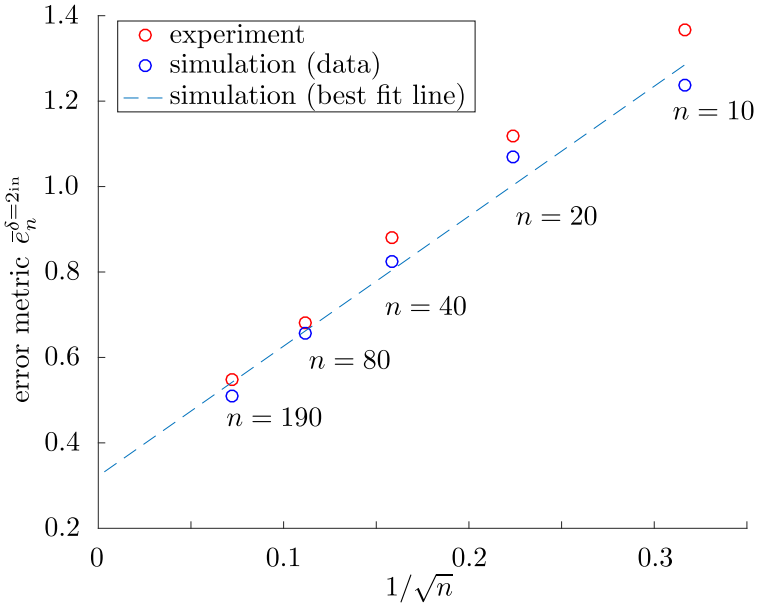


Figure 2.7: Relationship between the error metric e_δ and number of robots n ($\delta = 2$in, ring pattern). Note that as n increases, the error tends toward a nonzero value, which is due to the nonzero δ. Nonetheless, the decrease is quite linear with respect to $\frac{1}{\sqrt{n}}$, as expected.

2.5 Conclusion

In this chapter, we analyze a decentralized stochastic control law, which drives a robotic swarm towards a pre-defined target distribution. We also present a testbed experiment validation of the control law. Despite the significant differences between the real world platform and the theoretical simulation, we still achieve expected convergence at a practical time rate. The reason lies in the robustness of stochastic control, which is not sensitive to the inherit errors from the sensor, controller, etc. The error metric is experimentally justified to converge as $\frac{1}{\sqrt{N}}$, as hypothesized in Chapter 1. The result of our experiments suggests that this control law is an effective choice for distributing source-constraint robots with no localization or communication abilities.

CHAPTER 3

Analysis of Advection-Diffusion-Reaction System Modeling Spatial Coverage by Robotic Swarm

3.1 Introduction

In recent years, there has been a growing interest in the development of *robotic swarms* [BFB13] for a range of applications, including environmental sensing, exploration, mapping, disaster response, surveillance, cooperative manipulation, and even nanomedicine [TFL14]. Indeed, advances in manufacturing, computing, sensing, actuation, control, and other technologies have already enabled the development of a variety of low-cost robotic platforms that can be deployed in large numbers, e.g. [MCF13, CCD16, GDG16].

While the technology to create robotic swarms is progressing, it remains a challenge to predict and control these systems' collective behaviors when they operate in uncertain, unstructured, GPS-denied environments. Another constraint is that inter-robot communication may need to be minimized or excluded in order to conserve power and reduce the possibility of detection by adversaries. Importantly, control policies and verification methods for robotic swarms must accommodate *non-deterministic behaviors* that arise in autonomous systems [Roa]. Stochasticity in robots' motion and decisions can arise from inherent sensor and actuator noise, especially in small, highly resource-restricted platforms. Stochasticity may also be intentionally introduced, for example when robots are programmed to perform random walks for probabilistic search and tracking missions [RL15], or to switch probabilistically between behavioral states or tasks in a manner similar to social insects. Social insect colonies provide a useful paradigm for robotic swarm control in that they display robust collective behaviors that emerge from the decentralized decisions of numerous individuals,

which act on locally perceived information [BDT99].

Control methodologies for robotic swarms should be scalable with the number of robots and reliant on limited human supervision, since situational awareness decreases with large robot populations. Toward this end, we employ a methodology that is based on models of the robots' decision-making and motion at multiple levels of abstraction. The multi-level modeling framework is adopted from the disciplines of stochastic chemical kinetics and fluid dynamics, and it has been used by others [PCM11, HW08] to describe the population dynamics of large numbers of robots. This framework has also been used to model collective behaviors in biological swarms, such as flocking, schooling, chemotaxis, pattern formation, and predator-prey interactions [OL13].

In our modeling framework, the *microscopic model* is a discrete model that represents the actions of individual robots. We consider swarms of robots that display stochastic motion and decision-making as described above, while also moving according to a programmed deterministic velocity field. Each robot's stochastic movement can be modeled as a Brownian motion with an associated diffusion coefficient. Since the motion of each robot consists of a deterministic advection and a stochastic Brownian walk, it is governed by a stochastic differential equation (SDE). A robot's stochastic transition between two behavioral states can be modeled as a chemical reaction with a programmable transition probability rate.

Implementations of the microscopic model can be computationally expensive to simulate, requiring exhaustive parametric studies, and intractable for analysis as the number of robots increases. To overcome these limitations, the microscopic model can be abstracted to a lower-dimensional continuum representation, the *macroscopic model*, which consists of a set of advection-diffusion-reaction (ADR) partial differential equations (PDEs). These equations govern the spatiotemporal dynamics of *density fields* of robots in different behavioral states. The macroscopic model enables a quantitative characterization of population behaviors, since it is amenable to analytical treatment and numerical experiments. In addition, techniques for control and optimization of PDEs can be applied to compute values of the model parameters that produce a desired global objective. These parameters define the robots' programmable control policies for motion and state transitions, and the resulting collective behavior of the

robots follows the macroscopic model prediction in expectation. Scalability of this "top-down" control approach is ensured by the fact that the dimensionality of the macroscopic model is independent of the number of robots. Human supervisory control can be exercised in the specification of the global objective and the set of tunable model parameters and state transitions.

In recent years, there have been various applications of control-theoretic techniques to PDE macroscopic models of multi-agent systems for the purpose of synthesizing agent controllers that produce desired collective behaviors. ADR PDE models in particular have been proposed to design robot control policies that achieve target spatial distributions of robot activity over a bounded domain [EB15] and that drive the swarm to a distribution that is proportional to a locally measured scalar field [EAB16]. ADR PDEs have also been used to control the probability density functions of multi-dimensional stochastic processes [AB13], develop multi-agent coverage and search strategies that are inspired by bacterial chemotaxis [MH12], and maximize the probability of swarm robotic presence in a desired region [ML06]. Other work on PDE-based analysis and design of agent control laws includes a study of multi-agent consensus protocols in an Eulerian framework [CFT08]; strategies for confining a population of agents, represented as a continuum, with a few discrete leader agents [CP12]; and an approach to flocking control for a group of agents governed by the kinetic Cucker-Smale model [PRT15].

The literature above addresses the problem of designing the rules that govern robots' behaviors and decisions. However, there has been relatively little effort toward a principled approach to determining the *required number of robots* and *optimal robot specifications*, such as sensing and communication ranges, for a desired collective task. An impediment to developing such an approach is the absence of a rigorous and generalizable analysis of the correspondence between continuum and discrete models of a swarm [BV15]. Recent work on mean field games [LL07, BFY13, FS14] demonstrates the convergence of optimal controls of a large number of agents to optimal controls of a mean-field limit system. However, the work does not analyze the convergence of the agent-based model to the mean-field model for a fixed set of controls.

In this chapter, we address this challenge for robotic swarms that can be modeled as ADR PDEs at the macroscopic level. We derive a rigorous error bound on the discrepancy between the microscopic and macroscopic models, which depends on the swarm population size (alternatively, the number of swarm deployments), the robot sensing radius length, and the time discretization of the microscopic and macroscopic models. Our derivation employs a representation of each robot as a circular "blob function" [CB16, MB02] with a small parameter that represents the robot's maximum sensing radius. We formulate the discrete density functions of robots in different states and robots' cumulative activity over the domain by summing all of the corresponding blobs. We show that as the number of robots approaches infinity, the discrete density functions converge to the continuous solution of the macroscopic model. We illustrate our approach for a simulated scenario in which a swarm of micro-aerial vehicles must pollinate a crop field, similar to the problem in [EB15]. We apply the optimal control approach in [EB15] to compute vehicle control policies that achieve a target spatial distribution of pollination. We also use our derived error bound to estimate the required swarm size that will achieve the target pollination distribution within a specified percentage of accuracy. In addition, we demonstrate the effect of the maximum sensing radius on the swarm performance and show that an optimal radius length exists for a given swarm size. Notably, the analysis performed here can also be applied to other stochastic control strategies for robotic swarms, such as [PCM11, HW08, EAB16].

In summary, the novelty of this chapter is two-fold:

1. We provide a rigorous analysis of the error bound between the aforementioned microscopic and macroscopic models, which is still absent in the literature on stochastic control of multi-agent systems with state transitions. This analysis, together with our optimal control approach in [EB15] which approximates the target distribution using the macroscopic model, provides a formal mathematical validation of our swarm control strategy.

2. Based on the scaling laws that are observed in the error estimates, we propose a principled approach to determine the *required number of robots* and *optimal robot sensing*

radius that will achieve a target distribution within a specified error.

This chapter is organized as follows. Section 3.2 describes our task objective and the robot capabilities and behaviors, and Section 3.3 outlines our design procedure for computing the number of robots and the robots' sensing radius, velocity, and pollination rates. Section 3.4 defines the microscopic model, the blob function, and the actual density fields of robots and their pollination activity, and Section 3.5 formulates the macroscopic model, an operator splitting method for numerically solving this model, and the expected density fields. Section 3.6 summarizes our optimal control approach, first presented in [EB15], to designing robot control policies for target spatial coverage. In Section 3.7, we provide our convergence analysis of the estimated error between the actual, expected, and target density fields. We validate our analysis and design procedure with simulations in Section 3.8 and conclude in Section 3.9.

3.2 Task Objective

In this section, we present the task objective of the robot control scenario defined in [EB15], which is the basis of the analysis in this chapter. We consider a crop field $\Omega \in \mathbb{R}^2$ with several rows of flowers to be pollinated by a swarm of N micro-aerial vehicles. There are n_f types of crops in the field, and $\Gamma_j \subset \Omega$ denotes the region of the field that is occupied by crops of type $j \in \{1, ..., n_f\}$. The **task objective**, which must be completed within time T, is to achieve a spatial distribution of pollination activity over the field within a specified error γ_d relative to a target pollination distribution $\rho_\Omega(\boldsymbol{x})$, where $\boldsymbol{x} \in \Omega$.

The swarm originates from a location in the field called the *hive*. The robots are assumed to have sufficient power to undertake brief flights from the hive, and they return to the hive to recharge after a complete flight. Each robot is equipped with a compass and thus can fly with a specified heading. However, the robots' stringent power constraints make it infeasible for them to use inter-robot communication or GPS sensors for global localization. A computer in the hive serves as a supervisory agent and calculates the parameters of the robots' motion and state transitions prior to their flight.

Each robot $i \in \{1, ..., N\}$ performs the following actions during a flight. Upon deploying from the hive, each robot flies with a combination of a time-dependent velocity field $\boldsymbol{v}(t) \in \mathbb{R}^2$ and a Brownian motion, which is characterized by a diffusion coefficient parameter $D > 0$. We assume that the flowers are distributed densely enough such that a robot can always detect at least one flower within its sensing radius δ when it flies over the crop rows. The sensing radius can be adjusted within a maximum radius, which is determined by the capability of the robot. When a robot is flying over crops of type j, it decides with a time-dependent probability per unit time $k_j(t)$, the *pollination rate*, to pause at a flower within its sensing range and hover for pollination. The robot resumes flying with a fixed probability per unit time k_f, which determines the time taken to pollinate.

3.3 Design Procedure for Target Performance Bounds

Here we present a procedure for computing the number of robots N, the robot velocity $\boldsymbol{v}(t)$, and the robot pollination rates $k_j(t)$ and selecting the robot sensing radius δ to achieve the task objective defined in Section 3.2. The details of certain steps in the procedure are given in subsequent sections, as referenced below. We illustrate this computational procedure in Section 3.8 for an example pollination scenario.

1. Set values of the parameters n_f, Γ_j, T, $\rho_\Omega(\boldsymbol{x})$, γ_d, D, and k_f, defined in Section 3.2, and Δt, $\boldsymbol{X}_0$, defined in Section 3.4.

2. Compute the robot control parameters $\boldsymbol{v}(t)$ and $k_j(t)$, defined in Section 3.2, by applying the optimal control technique described in Section 3.6 to the macroscopic model, defined in Section 3.5.

3. Choose a value of δ and two values of N. Simulate the microscopic model, defined in Section 3.4, for each value of N with the chosen δ and the computed control parameters $\boldsymbol{v}(t)$ and $k_j(t)$.

4. For each value of N, compare the actual distribution of pollination in the microscopic

model to the target distribution $\rho_\Omega(\boldsymbol{x})$ and compute the discrepancy between them.

5. Use the convergence analysis in Section 3.7 to estimate the required N such that the discrepancy is less than γ_d.

6. Simulate the microscopic model for several values of δ with the estimate of the required N, and select the δ that yields the minimum discrepancy.

3.4 Microscopic Model

3.4.1 Robot controller

We use the same robot controller as in our previous work [EB15]. We discretize the time span of swarm deployment $[0, T]$ into M equal time steps:

$$0 = t_0 < t_1 < ... < t_M = T, \quad t_m = m\Delta t. \tag{3.1}$$

The controller that drives each robot is illustrated by the state-transition diagram in Fig. 3.1. Robots switch stochastically between two states, *Flying* and *Hovering*. We define the index sets of robots in each state at time t_m:

$$F_m = \{i : \text{Robot } i \text{ is } \textit{Flying}\},$$
$$H_{j,m} = \{i : \text{Robot } i \text{ is } \textit{Hovering} \text{ over crops of type } j\}.$$

All of the robots start from $\boldsymbol{X}_0 \in \Omega$ in the *Flying* state. At the start of each time step Δt, each *Flying* robot that is over crops of type j switches to *Hovering* at a flower with probability $k_j(t_m)\Delta t$, and each *Hovering* robot returns to *Flying* with probability $k_f \Delta t$. We choose Δt to be small enough such that $k_j(t_m)\Delta t \leq 1$ and $k_f \Delta t \leq 1$, since these probabilities can at most be 1. The robots' state transitions can be modeled as the following reactions, where $\phi \in [0, 1]$ is a uniformly distributed random number:

$$\begin{aligned} i \in F_m &\xrightarrow{\text{if } \phi \leq k_j(t_m)\Delta t} i \in H_{j,m+1} \\ i \in H_{j,m} &\xrightarrow{\text{if } \phi \leq k_f \Delta t} i \in F_{m+1} \end{aligned} \tag{3.2}$$

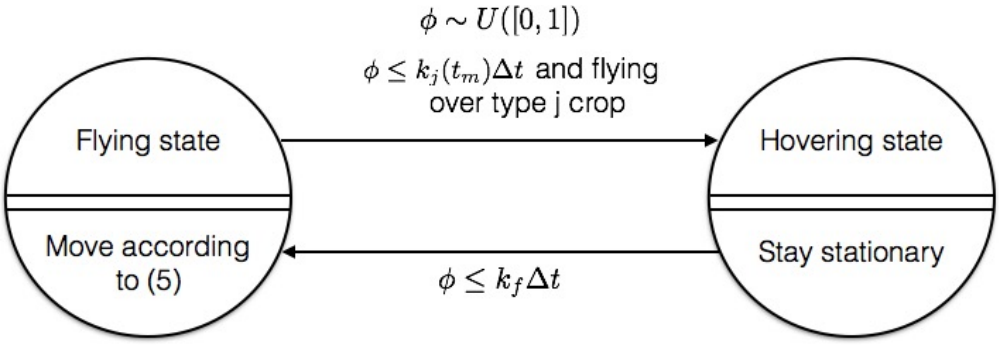


Figure 3.1: State-transition diagram of robot controller for pollination. The diagram outlines a program that would run on a single robot. [ZBE18] Copyright ©2017 IEEE.

After generating ϕ and switching states if ϕ satisfies the condition associated with its possible reaction in (3.2), each robot executes the motion controller that is defined for its current state over the duration Δt. We define the domain as unbounded, and robots may exit and re-enter the bounded subregion of the domain that represents the crop field. The position of robot i at time t_m, the beginning of the time step, is denoted by $\boldsymbol{X}_m^i \in \mathbb{R}^2$. Each *Hovering* robot stays at the location of the flower that it is pollinating, i.e.

$$\boldsymbol{X}_{m+1}^i = \boldsymbol{X}_m^i \quad \forall i \in \cup_{j=1}^{n_f} H_{j,m+1}. \tag{3.3}$$

Each *Flying* robot moves according to the stochastic differential equation,

$$\boldsymbol{dX}(t) = \boldsymbol{v}(t)\Delta t + \sqrt{2D}\boldsymbol{dB}(t), \tag{3.4}$$

where $\boldsymbol{B}(t)$ is the standard Brownian motion. We simulate this motion using a first-order discretization of Eq. (3.4),

$$\boldsymbol{X}_{m+1}^i = \boldsymbol{X}_m^i + \boldsymbol{v}(t_m)\Delta t + \sqrt{2D\Delta t}\Delta \boldsymbol{Z}_m^i \quad \forall i \in F_{m+1}, \tag{3.5}$$

where $\Delta \boldsymbol{Z}_m^i$ are independent, normally distributed random variables with zero mean and unit variance in $\mathbb{R}^2$.

3.4.2 Density fields of robots and pollination activity

In this section, we define the density fields of the microscopic model. During a deployment, when a *Flying* robot switches to the *Hovering* state for pollination, it randomly selects a

flower that it identifies within its sensing radius δ. In order to compute the density fields of robots and their pollination activity, we model the probability density function of the location that the robot chooses to pollinate as a *blob function* $G_\delta(\boldsymbol{x})$. We define the blob function as:

$$G_\delta(\boldsymbol{x}) = \begin{cases} \dfrac{C_g}{\delta^2} \exp\left(\dfrac{1}{|\boldsymbol{x}|^2/\delta^2 - 1}\right) & \text{if } |\boldsymbol{x}| < \delta, \\ 0 & \text{otherwise,} \end{cases} \tag{3.6}$$

where $C_g \approx 2.1436$ so that

$$\int_{\mathbb{R}^2} G_\delta(\boldsymbol{x}) d\boldsymbol{x} = 1.$$

G_δ satisfies the following properties:

1. $G_\delta \in C_0^\infty(\mathbb{R}^2)$, and its support is $\{\boldsymbol{x} : |\boldsymbol{x}| \leq \delta\}$;

2. $\forall \boldsymbol{x}$, $|G_\delta(\boldsymbol{x})| \leq C_g e^{-1} \delta^{-2} < \delta^{-2}$;

3. $\forall \boldsymbol{x}$, $|\partial_{x_i} G_\delta(\boldsymbol{x})| = \mathcal{O}(\delta^{-3})$ and $|\partial_{x_i x_j} G_\delta(\boldsymbol{x})| = \mathcal{O}(\delta^{-4})$, $i, j = 1, 2$.

Fig. 3.2 illustrates a domain with three blob functions, each with the same sensing radius parameter δ.

Remark 3.4.1. *The blob function serves as a mean field approximation of pollination activity. That is, instead of modeling a robot's selection of a particular flower to pollinate, we consider the probability density of the robot's flower visits over multiple deployments, or alternatively, the flower visits by a large number of robots over a single deployment. Each crop row is modeled as a continuum of possible pollination locations, and thus a robot can choose to hover at any position within the support of its corresponding blob function.*

For all $\Sigma \subset \mathbb{R}^2$, we define the indicator function as

$$\mathbb{1}_\Sigma(\boldsymbol{x}) = \begin{cases} 1, \text{ if } \boldsymbol{x} \in \Sigma \\ 0, \text{ otherwise.} \end{cases}$$

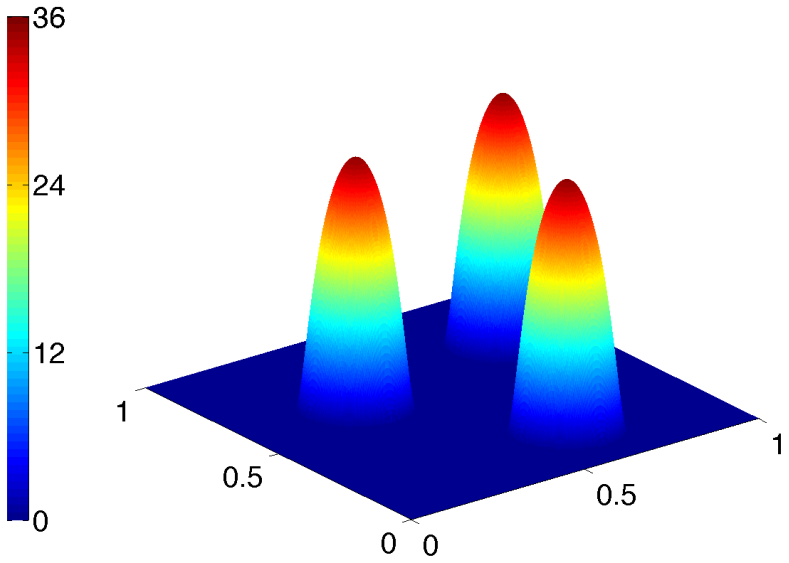


Figure 3.2: A domain with three blob functions $G_\delta(\boldsymbol{x})$, where $\delta = 0.15$. [ZBE18] Copyright ©2017 IEEE.

We also define

$$dis(\boldsymbol{x}, \Sigma) := \inf\{|\boldsymbol{x} - \boldsymbol{y}| : \boldsymbol{y} \in \Sigma\},$$

$$\Sigma_{in}^{\zeta} := \{\boldsymbol{x} : dis(\boldsymbol{x}, \Sigma^c) \geq \zeta\}, \tag{3.7}$$

$$\Sigma_{out}^{\zeta} := \{\boldsymbol{x} : dis(\boldsymbol{x}, \Sigma) \leq \zeta\}, \tag{3.8}$$

for some constant $\zeta > 0$, where Σ^c is the complement of Σ. From these definitions, Σ_{in}^{ζ} and Σ_{out}^{ζ} are obtained by shrinking and expanding, respectively, the boundary of Σ by a layer of width ζ. Hence, $\Sigma_{in}^{\zeta} \subset \Sigma \subset \Sigma_{out}^{\zeta}$.

Let $\boldsymbol{X}(t)$ be a stochastic process in $\mathbb{R}^2$ that satisfies the SDE (3.4). For all $t > s \geq 0$ and $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^2$, we denote the transition probability measure by $P(\Sigma, t|\boldsymbol{y}, s) = P(\boldsymbol{X}(t) \in \Sigma|\boldsymbol{X}(s) = \boldsymbol{y})$ and the transition probability density function (pdf) by $p_e(\boldsymbol{x}, t|\boldsymbol{y}, s)$. These functions satisfy

$$P(\Sigma, t|\boldsymbol{y}, s) = \int_{\Sigma} p_e(\boldsymbol{x}, t|\boldsymbol{y}, s) d\boldsymbol{x},$$

$$p_e(\boldsymbol{x}, t|\boldsymbol{y}, s) = \frac{1}{4\pi D(t-s)} \exp\left\{-\frac{|\boldsymbol{x} - \boldsymbol{y} - \int_s^t \boldsymbol{v}(\tau) d\tau|^2}{4D(t-s)}\right\}.$$

In our simulation of the microscopic model, we discretize the velocity $\boldsymbol{v}(t)$, and hence the transition pdf from time t_m to time t_{m+1} is given by

$$p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m) = \frac{1}{4\pi D\Delta t} \exp\left\{-\frac{|\boldsymbol{x}-\boldsymbol{y}-\boldsymbol{v}(t_m)\Delta t|^2}{4D\Delta t}\right\}. \tag{3.9}$$

We now define the *actual density fields* of *Flying* robots and *Hovering* robots, respectively, at each location $\boldsymbol{x} \in \Omega$ and each time t_m as:

$$\rho_1^\delta(\boldsymbol{x}, t_m) := \frac{1}{N} \sum_{i \in F_m} G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i), \tag{3.10}$$

$$\rho_2^\delta(\boldsymbol{x}, t_m) := \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in H_{j,m}} G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i). \tag{3.11}$$

To confirm that these are robot density fields, note that

$$\int_\Sigma G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i)\boldsymbol{dx} \;\approx\; \mathbb{1}_\Sigma(\boldsymbol{X}_m^i).$$

The above identity strictly holds only when $\boldsymbol{X}_m^i \notin \Sigma_{out}^\delta - \Sigma_{in}^\delta$; otherwise, the range of the blob will exceed the boundary and will cause *coverage outflow*, an error introduced in Section 3.7. Hence

$$\int_\Sigma \rho_1^\delta(\boldsymbol{x}, t_m)\boldsymbol{dx} \approx \frac{1}{N} \sum_{i \in F_m} \mathbb{1}_\Sigma(\boldsymbol{X}_m^i), \tag{3.12}$$

$$\int_\Sigma \rho_2^\delta(\boldsymbol{x}, t_m)\boldsymbol{dx} \approx \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in H_{j,m}} \mathbb{1}_\Sigma(\boldsymbol{X}_m^i), \tag{3.13}$$

which are the numbers of *Flying* and *Hovering* robots, respectively, that are in region Σ at time t_m, divided by N.

We also define the actual density fields of *Flying* and *Hovering* robots, respectively, that are present after robots execute state transitions according to reactions (3.2) but before they execute their motion controllers during the time step Δt:

$$\begin{aligned} \bar{\rho}_1^\delta(\boldsymbol{x}, t_m) &:= \frac{1}{N} \sum_{i \in F_{m+1}} G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i), \\ \bar{\rho}_2^\delta(\boldsymbol{x}, t_m) &:= \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in H_{j,m+1}} G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i). \end{aligned} \tag{3.14}$$

Note that by Eq. (3.3), the positions of *Hovering* robots is unchanged during the time step, and therefore

$$\bar{\rho}_2^{\delta}(\boldsymbol{x}, t_m) = \rho_2^{\delta}(\boldsymbol{x}, t_{m+1}). \tag{3.15}$$

We denote the density of robot state transitions from *Flying* to *Hovering* between times t_m and t_{m+1} as $FTH(\boldsymbol{x}, t_m)$, and the density of transitions from *Hovering* to *Flying* as $HTF(\boldsymbol{x}, t_m)$. These densities can be expressed as:

$$\begin{aligned} FTH(\boldsymbol{x}, t_m) &:= \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in F_m} I_{i,j,m} L_j(\boldsymbol{X}_m^i) G_{\delta}(\boldsymbol{x} - \boldsymbol{X}_m^i), \\ HTF(\boldsymbol{x}, t_m) &:= \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in H_{j,m}} J_{i,j,m} G_{\delta}(\boldsymbol{x} - \boldsymbol{X}_m^i), \end{aligned}$$

where $L_j(\boldsymbol{x}) = \mathbb{1}_{\Gamma_j}(\boldsymbol{x})$, and $I_{i,j,m},\ J_{i,j,m}$ are independent random variables with

$$\begin{aligned} P(I_{i,j,m} = 1) = \Delta t \cdot k_{j,m}, \quad P(I_{i,j,m} = 0) = 1 - \Delta t \cdot k_{j,m}, \\ P(J_{i,j,m} = 1) = \Delta t \cdot k_f, \quad P(J_{i,j,m} = 0) = 1 - \Delta t \cdot k_f, \end{aligned}$$

with indices $i = 1, ..., N$, $j = 1, ..., n_f$, and $m = 1, ..., M$ and $k_{j,m} := k_j(t_m)$. According to the reaction network (3.2),

$$\begin{aligned} \bar{\rho}_1^{\delta}(\boldsymbol{x}, t_m) &= \rho_1^{\delta}(\boldsymbol{x}, t_m) - FTH(\boldsymbol{x}, t_m) + HTF(\boldsymbol{x}, t_m), \\ \bar{\rho}_2^{\delta}(\boldsymbol{x}, t_m) &= \rho_2^{\delta}(\boldsymbol{x}, t_m) + FTH(\boldsymbol{x}, t_m) - HTF(\boldsymbol{x}, t_m). \end{aligned} \tag{3.16}$$

At each time t_m, the total number of state transitions from *Flying* to *Hovering* in the region Σ is given by:

$$\int_{\Sigma} FTH(\boldsymbol{x}, t_m) d\boldsymbol{x} \approx \frac{1}{N} \sum_{j=1}^{n_f} \sum_{i \in F_m} I_{i,j,m} L_j(\boldsymbol{X}_m^i) \mathbb{1}_{\Sigma}(\boldsymbol{X}_m^i). \tag{3.17}$$

Since each transition from *Flying* to *Hovering* indicates a robot pollination visit, $FTH(\boldsymbol{x}, t_m)$ is also the *actual density field of pollination activity* at time t_m. Thus, the actual cumulative density field of pollination activity by the swarm from time 0 to time t_m is given by:

$$\rho_3^{\delta}(\boldsymbol{x}, t_m) = \sum_{\tau=0}^{m-1} FTH(\boldsymbol{x}, t_\tau). \tag{3.18}$$

We define the tuple of actual density fields as

$$\boldsymbol{\rho}^{\delta}(\boldsymbol{x}, t_m) = \left(\rho_1^{\delta}(\boldsymbol{x}, t_m), \rho_2^{\delta}(\boldsymbol{x}, t_m), \rho_3^{\delta}(\boldsymbol{x}, t_m)\right). \tag{3.19}$$

The goal of our analysis is to compare ρ_3^{δ} to the expected density field of pollination, which is defined in the next section.

3.5 Macroscopic Model

3.5.1 Definition

The macroscopic model consists of a set of advection-diffusion-reaction (ADR) PDEs that describe the time evolution of the expected spatial distribution of the swarm. The model presented here was first defined in [EB15] for a similar pollination scenario. The states of the macroscopic model are $\rho_1(\boldsymbol{x}, t)$, $\rho_2(\boldsymbol{x}, t)$, and $\rho_3(\boldsymbol{x}, t)$, the *expected density fields* of *Flying* robots, *Hovering* robots, and cumulative pollination from time 0 to t, respectively. Using the parameters $\boldsymbol{v}(t)$, $k_j(t), j = 1, 2, ..., n_f$, k_f, D, and L_j that are defined in the microscopic model, the macroscopic model is given by

$$\begin{cases} \dfrac{\partial \rho_1}{\partial t} = -\boldsymbol{v} \cdot \nabla \rho_1 + D\Delta\rho_1 - \displaystyle\sum_{j=1}^{n_f} k_j L_j \rho_1 + k_f \rho_2, \\ \dfrac{\partial \rho_2}{\partial t} = \displaystyle\sum_{i=1}^{n_f} k_j L_j \rho_1 - k_f \rho_2, \\ \dfrac{\partial \rho_3}{\partial t} = \displaystyle\sum_{i=1}^{n_f} k_j L_j \rho_1, \end{cases} \tag{3.20}$$

with initial conditions specifying that all robots start in the *Flying* state and are distributed according to a blob function centered at $\boldsymbol{X}_0$:

$$\rho_1(\boldsymbol{x}, 0) = G_{\delta}(\boldsymbol{x} - \boldsymbol{X}_0),\ \rho_2(\boldsymbol{x}, 0) = 0,\ \rho_3(\boldsymbol{x}, 0) = 0. \tag{3.21}$$

The initial conditions of the macroscopic model and microscopic model are consistent, i.e.

$$\rho_i(\boldsymbol{x}, 0) = \rho_i^{\delta}(\boldsymbol{x}, 0), \quad i = 1, 2, 3. \tag{3.22}$$

We define the tuple of expected density fields as

$$\boldsymbol{\rho}(\boldsymbol{x},t) = (\rho_1(\boldsymbol{x},t), \rho_2(\boldsymbol{x},t), \rho_3(\boldsymbol{x},t)). \tag{3.23}$$

3.5.2 Numerical solution

We use the operator splitting method to numerically solve the macroscopic model with the same time discretization as in Eq. (3.1). We define the following three operators:

$$\begin{aligned} ADV_m(\boldsymbol{\rho}) &= (\ -\boldsymbol{v}(t_m)\cdot\nabla\rho_1\ ,\ 0\ ,\ 0\ ), \\ DIFF(\boldsymbol{\rho}) &= (\ D\Delta\rho_1\ ,\ 0\ ,\ 0\ ), \\ REACT_m(\boldsymbol{\rho}) &= \left(k_f\rho_2 - \sum_{j=1}^{n_f} k_{j,m}L_j\rho_1,\ \sum_{j=1}^{n_f} k_{j,m}L_j\rho_1 - k_f\rho_2,\ \sum_{j=1}^{n_f} k_{j,m}L_j\rho_1\right). \end{aligned} \tag{3.24}$$

We split the macroscopic model (3.20), (3.21) into three parts:

$$\frac{\partial\boldsymbol{\rho}}{\partial t} = ADV_m(\boldsymbol{\rho}), \tag{3.25}$$

$$\frac{\partial\boldsymbol{\rho}}{\partial t} = DIFF(\boldsymbol{\rho}), \tag{3.26}$$

$$\frac{\partial\boldsymbol{\rho}}{\partial t} = REACT_m(\boldsymbol{\rho}). \tag{3.27}$$

Denote the solution operators of Eq. (3.25), (3.26), and (3.27) with respect to time step Δt by $H_1(\Delta t)$, $H_2(\Delta t)$, and $H_3(\Delta t)$, respectively. That is, $\boldsymbol{\rho}(\boldsymbol{x},t_{m+1}) = H_1(\Delta t)\boldsymbol{\rho}(\boldsymbol{x},t_m)$ if $\boldsymbol{\rho}(\boldsymbol{x},t_{m+1})$ is the solution of Eq. (3.25) with initial condition $\boldsymbol{\rho}_0(\boldsymbol{x}) = \boldsymbol{\rho}(\boldsymbol{x},t_m)$.

Using these operators, we can compute the expected density fields at time t_{m+1} as

$$\boldsymbol{\rho}(\boldsymbol{x},t_{m+1}) = H_1(\Delta t)H_2(\Delta t)H_3(\Delta t)\boldsymbol{\rho}(\boldsymbol{x},t_m). \tag{3.28}$$

This is a first-order splitting method, i.e.

$$\int_{\mathbb{R}^2} |\boldsymbol{\rho}_e(\boldsymbol{x},T) - \boldsymbol{\rho}(\boldsymbol{x},T)|d\boldsymbol{x}\ \leq\ C_\delta\Delta t, \tag{3.29}$$

where $\boldsymbol{\rho}_e$ is the exact solution of model (3.20) and $\boldsymbol{\rho}$ is defined by Eq. (3.28). We note that C_δ depends on δ and that $\lim_{\delta\to 0} C_\delta = \infty$. We choose the values of Δt and δ based on the numerical simulation results in Section 3.8 to ensure that the above error is small.

We also define the expected density fields of *Flying* and *Hovering* robots that are present after the reactions but before robot motion during a time step,

$$\bar{\rho}_i(\boldsymbol{x}, t_m) := H_3(\Delta t)\rho_i(\boldsymbol{x}, t_m),\ i = 1, 2, \tag{3.30}$$

which correspond to $\bar{\rho}_i^{\delta}(\boldsymbol{x}, t_m),\ i = 1, 2$ in the microscopic model. Equations (3.28) and (3.30) also yield

$$\rho_i(\boldsymbol{x}, t_{m+1}) = H_1(\Delta t)H_2(\Delta t)\bar{\rho}_i(\boldsymbol{x}, t_m),\ i = 1, 2. \tag{3.31}$$

From the definitions of $H_1(\Delta t)$ and $H_2(\Delta t)$, we have that

$$\rho_1(\boldsymbol{x}, t_{m+1}) = \int_{\mathbb{R}^2} \bar{\rho}_1(\boldsymbol{y}, t_m)p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m)\boldsymbol{dy}, \tag{3.32}$$

$$\rho_2(\boldsymbol{x}, t_{m+1}) = \bar{\rho}_2(\boldsymbol{x}, t_m). \tag{3.33}$$

Equation (3.32) holds because the transition pdf $p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m)$ is also the Green's function of the advection-diffusion equation, see theorem 1.2.2 in Chapter 1, or Theorem 2.1 in [Pav14] for a one-dimensional case.

We numerically solve the three operators over a square domain $\bar{\Omega}$ with Neumann boundary conditions. The domain is defined to be large enough to contain all the robots almost surely over the entire duration of the deployment. For the advection operator (3.25), we use the following Lax-Friedrichs scheme.

$$\frac{\rho_{mn}^{j+1} - \frac{1}{4}(\rho_{m+1,n}^{j} + \rho_{m-1,n}^{j} + \rho_{m,n+1}^{j} + \rho_{m,n-1}^{j})}{\Delta t} = \frac{(v\rho)_{m-1,n}^{j} - (v\rho)_{m+1,n}^{j}}{2h} + \frac{(v\rho)_{m,n-1}^{j} - (v\rho)_{m,n+1}^{j}}{2h}$$

with CFL condition:

$$\frac{\Delta t}{h} \leq \frac{1}{\|v\|_{\infty}}.$$

For the diffusion operator (3.26), we use the Crank-Nicolson scheme

$$\frac{(\rho_{mn}^{j+1} - \rho_{mn}^{j})}{D\Delta t} = \frac{\left(\rho_{m+1,n}^{j} - 2\rho_{m,n}^{j} + \rho_{m-1,n}^{j}\right) + \left(\rho_{m,n+1}^{j} - 2\rho_{m,n}^{j} + \rho_{m,n-1}^{j}\right)}{2h^2} + \frac{\left(\rho_{m+1,n}^{j+1} - 2\rho_{m,n}^{j+1} + \rho_{m-1,n}^{j+1}\right) + \left(\rho_{m,n+1}^{j+1} - 2\rho_{m,n}^{j+1} + \rho_{m,n-1}^{j+1}\right)}{2h^2}. \tag{3.34}$$

Method (3.34) is semi-implicit. To avoid solving a large scale linear system of equations, we apply the discrete cosine transform

$$\hat{\rho}^j_{k,l} = \sum_{m=1}^{M}\sum_{n=1}^{N} \rho^j_{mn} \sin\frac{k\pi m}{M+1}\sin\frac{l\pi n}{N+1},$$

so that (3.34) turns into the explicit form

$$\hat{\rho}^{j+1}_{k,l} = \frac{1+\lambda_{k,l}}{1-\lambda_{k,l}}\hat{\rho}^j_{k,l}$$

where

$$\lambda_{k,l} = \frac{D\cdot\Delta t}{2h^2}\left(2\cos\frac{k\pi}{M+1} + 2\cos\frac{l\pi}{N+1} - 4\right).$$

Then we apply the inverse discrete cosine transform to solve ρ from $\hat{\rho}$:

$$\rho^{j+1}_{m,n} = \frac{2}{M+1}\frac{2}{N+1}\sum_{k=1}^{M}\sum_{l=1}^{N}\hat{\rho}^{j+1}_{kl}\sin\frac{m\pi k}{M+1}\sin\frac{n\pi l}{N+1}.$$

Lastly, we solve the reaction operator (3.27) using the forward Euler scheme:

$$\frac{(\boldsymbol{\rho}^{j+1}_{mn} - \boldsymbol{\rho}^j_{mn})}{\Delta t} = REACT_m(\boldsymbol{\rho})|^j_{mn}\,.$$

3.6 Optimal Control of Coverage Strategies

We briefly summarize the optimal control problem that is solved in our previous work [EB15]. We use this approach to compute the optimal robot velocity $\mathbf{v}(t) = [v_1(t)\ v_2(t)]^T$ and pollination rates $k_j(t)$, $j = 1, ..., n_f$ that minimize the error between a target distribution ρ_Ω and the expected pollination field ρ_3 at a given time T. Note that the performance of the optimal control method is not the focus of this chapter.

For an open subset $X \subseteq \mathbb{R}^2$, $L^2(X)$ refers to the space of real-valued, square-integrable functions. The norm $\|\cdot\|_{L^2(X)}$ is defined as $\|f\|_{L^2(X)} = \left(\int_X |f(\boldsymbol{x})|^2 d\boldsymbol{x}\right)^{1/2}$ for each $f \in L^2(X)$. The notation $\left\langle\cdot,\cdot\right\rangle_{L^2(X)}$ refers to the inner product on $L^2(X)$, defined as $\int_X f(\boldsymbol{x})g(\boldsymbol{x})d\boldsymbol{x}$ for each $f, g \in L^2(X)$. For a natural number m, $\|\cdot\|_{L^2(X)^m}$ and $\left\langle\cdot,\cdot\right\rangle_{L^2(X)^m}$ refer to the natural extension of the norm and inner product on the product space $L^2(X)^m$. The vector of control parameters is defined as

$$\boldsymbol{u} := (u_1, u_2, ..., u_{n_f+2}),$$

where $u_1 = v_1$, $u_2 = v_2$, and $u_{j+2} = k_j$ for $j = 1, ..., n_f$. Then the optimal control problem is the following:

$$\min_{(\boldsymbol{\rho},\boldsymbol{u})\in Y\times U_{ad}} J(\boldsymbol{\rho},\boldsymbol{u}) \;=\; \frac{1}{2}\|\rho_3(\cdot,T)-\rho_\Omega\|^2_{L^2(\mathbb{R}^2)} + \frac{\lambda}{2}\|\boldsymbol{u}\|^2_{L^2(0,T)^\kappa}\;,\; \kappa = n_f+2 \tag{3.35}$$

subject to Eq. (3.20), (3.21). Hence, this is a PDE-constrained optimization problem. Here, $Y = C([0,T], L^2(\mathbb{R}^2)^3)$ is the space of vector-valued continuous functions $f : [0,T] \to L^2(\mathbb{R}^2)^3$, and U_{ad} is the set of admissible control inputs given by

$$U_{ad} = \{\boldsymbol{u} \in L^2(0,T)^{n_f+2} : u_i^{min} \le u_i(t) \le u_i^{max} \;\; \forall t \in (0,T)\},$$

where u_i^{min} and u_i^{max} are real-valued scalars defining the lower and upper bounds on the control parameters. These bounds are determined by the physical limitations on the robots, such as their maximum velocity. The bounds on the pollination rates k_j, $j = 1, ..., n_f$, additionally depend on the time step Δt, according to the constraint $k_j(t)\Delta t \le 1$.

The necessary conditions for optimality are used to derive a gradient descent method for numerically computing the optimal robot control parameters. We consider a reduced objective functional $\hat{J}$ corresponding to J in the optimal control problem (3.35). We define the following reduced problem:

$$\Xi : U_{ad} \to Y, \quad \min_{\boldsymbol{u}\in U_{ad}} \hat{J}(\boldsymbol{u}) := J(\Xi(\boldsymbol{u}),\boldsymbol{u}),$$

where Ξ is a control-to-state mapping which maps a control, $\boldsymbol{u}$, to $\boldsymbol{\rho}$, the corresponding solution of the macroscopic model (3.20), (3.21). The directional derivative of $\hat{J}$ is used in a gradient descent method to numerically compute the optimal robot control parameters. The expression for this derivative is given in the following claim, which is proved in [EB15].

Claim 3.6.1. *The reduced objective functional $\hat{J}$ is directionally differentiable along each $\boldsymbol{h} \in L^\infty(0,T)^{n_f+2}$, where $L^\infty(0,T)$ is the space of essentially bounded functions on the interval $(0,T)$. The directional derivative of $\hat{J}$ has the form*

$$d\hat{J}(\boldsymbol{u})\boldsymbol{h} = \int_0^T \langle \sum_{i=1}^{n_f+2} h_i B_i \rho, \boldsymbol{y}\rangle_{L^2(\mathbb{R}^2)^3} + \lambda\langle \boldsymbol{u},\boldsymbol{h}\rangle_{L^2(0,T)^{n_f+2}},$$

where $\boldsymbol{y}$ *is the solution of the backward-in-time adjoint equation*

$$
\begin{aligned}
-\frac{\partial y_1}{\partial t} &= \boldsymbol{v} \cdot \nabla y_1 + D\Delta y_1 + \sum_{j=1}^{n_f} k_j L_j(-y_1 + y_2 + y_3), \\
-\frac{\partial y_2}{\partial t} &= k_f y_1 - k_f y_2, \\
-\frac{\partial y_3}{\partial t} &= 0,
\end{aligned}
$$

with the final time condition

$$
y_1(\boldsymbol{x}, T) = y_2(\boldsymbol{x}, T) = 0,\ y_3(\boldsymbol{x}, T) = \rho_3(\boldsymbol{x}, T) - \rho_\Omega(\boldsymbol{x})
$$

and the input operators $\{B_i\}$ *defined as*

$$
B_1 = \begin{bmatrix} -\frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ B_2 = \begin{bmatrix} -\frac{\partial}{\partial x_2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},
$$

$$
B_i = \begin{bmatrix} -L_{i-2} & 0 & 0 \\ L_{i-2} & 0 & 0 \\ L_{i-2} & 0 & 0 \end{bmatrix}, \quad 3 \leq i \leq n_f + 2.
$$

The solution $\boldsymbol{y}$ of the above PDE plays the role of the covector in optimal control theory. However, a straightforward application of the maximum principle for finite-dimensional control systems to infinite-dimensional systems is not possible in general. Although there does exist a more general maximum principle for infinite-dimensional control systems such as those governed by PDEs [Fat99], this result is not applicable to our system due to the unboundedness of the control operators B_1 and B_2. An alternative approach to derive necessary conditions based on the first-order derivative of the control-to-state map is to use the Lagrange multiplier technique to formally derive the optimality conditions, and then rigorously prove the necessity of these conditions and the differentiability of the control-to-state map. This approach is outlined in [Tro10] and was applied in our prior work [EB15].

3.7 L^1-Convergence Analysis

In this section, we present the main result of this chapter: a rigorous convergence analysis to estimate the error between the expected density field $\boldsymbol{\rho}$ from the macroscopic model and the actual density field $\boldsymbol{\rho}^\delta$ from the microscopic model. Our result shows that the error depends on the number of robots N, the time discretization Δt, and the sensing radius δ.

In our analysis, we use the L^1 norm, which is the most natural norm for particle transportation, to quantify the degree of coverage by the swarm. This is because the L^1 norms of ρ_1^δ and ρ_2^δ directly measure the numbers of *Flying* robots and *Hovering* robots, respectively (see Eq. (3.12), (3.13)), and the L^1 norm of ρ_3^δ measures the cumulative number of crop visits (see Eq. (3.17), (3.18)), which is the metric of interest in the application. Note that in the optimal control method in Section 3.6, we use the L^2 norm in the objective function since it is convenient for optimal control. This is due to the inner-product structure of L^2 spaces, which makes them self-dual; L^1 function spaces lack this structure. Since our domain is a finite region, bounding the L^2 norm also bounds the L^1 norm according to the Cauchy-Schwarz inequality $\|\cdot\|_1 \leq C\|\cdot\|_2$.

The error bound that we derive in this section consists of four components: the *time-discretization error*, the *coverage outflow*, the *coverage insufficiency,* and the *sampling error*. The time-discretization error arises from our time-splitting method. The coverage outflow happens at the boundary of the region of crop rows $\Gamma_j, j = 1, 2, ..., n_f$: if a robot is pollinating in a row Γ_j at a position that is very close to the boundary of Γ_j, then part of the corresponding blob may exceed Γ_j, which generates some loss of coverage. Coverage insufficiency arises when there are too few robots in the swarm to cover the entire field, given the size of δ, and can be improved by deploying more robots. The most significant error component is the sampling error, which arises from the stochasticity in the robot motion and task switching. The error bound indicates the existence of an optimal δ for a fixed swarm size, which we verify in simulation in Section 3.8.

Let $\boldsymbol{\rho}^\delta(\boldsymbol{x}, t_m)$ and $\boldsymbol{\rho}(\boldsymbol{x}, t_m)$ be defined as in Eq. (3.19) and (3.28), respectively. We also

define the L^1 norm of a function $f : \mathbb{R}^2 \times [0,T] \to \mathbb{R}$ and the error functions as follows:

$$\|f(\cdot,t)\|_{1,\Sigma} := \int_\Sigma |f(\boldsymbol{x},t)|\boldsymbol{dx},\ \forall \Sigma \subset \mathbb{R}^2$$
$$e_i(\boldsymbol{x},t_m) := \rho_i(\boldsymbol{x},t_m) - \rho_i^\delta(\boldsymbol{x},t_m), \quad i = 1,2,3$$
$$E_m := \max\{\|e_1(\cdot,t_m)\|_1, \|e_2(\cdot,t_m)\|_1\}.$$

Theorem 3.7.1. *Assume that $\boldsymbol{v}(t) \in C^1([0,\infty])$, $D > 0$, and $k_i(t) \in C([0,\infty])$, $i = 1,...,n_f$. Suppose that $\bar{\Omega} \subset \mathbb{R}^2$ is a large enough square such that $\Omega \subset \bar{\Omega}$, and $\exists \zeta > 0$ such that*

$$\boldsymbol{X}_m^i \in \bar{\Omega}_{in}^{2\zeta} \quad \forall m = 1,...,M,\ i = 1,...,N$$

almost surely, and $\delta < \zeta$, $\Delta t \ll \zeta$. We define $|\bar{\Omega}|$ as the area of $\bar{\Omega}$, C as an independent constant, Γ as $\cup_{j=1}^{n_f}\Gamma_j$, and Γ_{in}^δ and Γ_{out}^δ as in Eq. (3.7) and (3.8). We also set

$$K = k_f + \sum_{j=1}^{n_f} \max_{t\in[0,T]} k_j(t),$$
$$P_\delta = \max_m P\left(\boldsymbol{X}_m^i \in \Gamma_{out}^\delta - \Gamma_{in}^\delta\right) = \mathcal{O}(\delta).$$

Then when N is sufficiently large, the following estimates are true with a probability greater than

$$1 - \frac{CT}{\Delta t} N^{\left[-\frac{1}{3}(\ln N)\Delta t^2 + 2\right]}\ . \tag{3.36}$$

(i) (Error in distributions of Hovering and Flying robots)

$$\|e_i(\cdot,t_m)\|_1 \le Ce^{KT}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right],\ i = 1,2 \tag{3.37}$$

uniformly in m.

(ii) (Error in distribution of cumulative pollination)

$$\|e_3(\cdot,t_m)\|_1 \le CKTe^{KT}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right] \tag{3.38}$$

uniformly in m.

Remark 3.7.2. *In the inequalities (3.37) and (3.38), the error terms are interpreted in the following way.*

1. $\frac{\ln N}{\sqrt{N}}$: *Sampling error;*

2. δ^{-4}: *Coverage insufficiency;*

3. P_δ: *Coverage outflow;*

4. Δt: *Time-discretization error.*

The sampling error is the main source of error in this model, due to the significant stochasticity in the robot motion and state transitions. The time-discretization error arises from the diffusion of the blob functions outside of $\bar{\Omega}$ *in the macroscopic PDE model. This error is not as significant as the other errors, since* $\bar{\Omega}$ *is chosen to be large enough to contain the entire swarm almost surely throughout the selected time span* $[0, T]$.

Note that the task-switching and the motion of the robots depend on each other, i.e. the motion depends on which state a robot is in, and the task-switching depends on whether a robot is above a crop region. We formulate the *error of motion* as

$$EM(\boldsymbol{x}) = \int_{\mathbb{R}^2} \bar{\rho}_1^\delta(\boldsymbol{y}, t_m) p(\boldsymbol{x}, t_{m+1} | \boldsymbol{y}, t_m) \boldsymbol{dy} - \rho_1^\delta(\boldsymbol{x}, t_{m+1}). \tag{3.39}$$

The first term is the expected density of *Flying* robots at t_{m+1} based on the actual density that is present after the reaction at t_m, and the second term is the actual density at t_{m+1}. We also formulate the *error of reaction* as

$$ER(\boldsymbol{x}) = -\Delta t \sum_{j=1}^{n_f} k_{j,m} L_j(\boldsymbol{x}) \rho_1^\delta(\boldsymbol{x}, t_m) + FTH(\boldsymbol{x}, t_m) + \Delta t k_f \rho_2^\delta(\boldsymbol{x}, t_m) - HTF(\boldsymbol{x}, t_m). \tag{3.40}$$

Here, $FTH(\boldsymbol{x}, t_m)$ and $HTF(\boldsymbol{x}, t_m)$ are the actual densities of robot state transitions between *Flying* and *Hovering*, whereas the other two terms are the expected densities of state transitions. To prove Theorem 3.7.1, we track the iteration of the error E_m over the time span $[0, T]$ using the following proposition.

Proposition 3.7.3 (Iteration of error)**.** *For all* $m = 0, ..., M-1$*, we have that*

$$E_{m+1} \ \leq \ (1 + \Delta t K) E_m + \|ER(\cdot)\|_1 + \|EM(\cdot)\|_1. \tag{3.41}$$

Proof of Proposition 3.7.3. We can decompose the error in the following way. First, using Eq. (3.32), we derive the inequality:

$$\begin{aligned} |e_1(\boldsymbol{x}, t_{m+1})| = & \ |\rho_1(\boldsymbol{x}, t_{m+1}) - \rho_1^\delta(\boldsymbol{x}, t_{m+1})| \\ = & \ \left| \int_{\mathbb{R}^2} \bar{\rho}_1(\boldsymbol{y}, t_m) p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m) d\boldsymbol{y} - \rho_1^\delta(\boldsymbol{x}, t_{m+1}) \right| \\ \leq & \ \left| \int_{\mathbb{R}^2} [\bar{\rho}_1(\boldsymbol{y}, t_m) - \bar{\rho}_1^\delta(\boldsymbol{y}, t_m)] p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m) d\boldsymbol{y} \right| + |EM(\boldsymbol{x})|. \end{aligned} \tag{3.42}$$

Then by taking the L^1 norm of both sides of Eq. (3.42), we obtain

$$\|e_1(\cdot, t_{m+1})\|_1 \leq \|\bar{\rho}_1(\cdot, t_m) - \bar{\rho}_1^\delta(\cdot, t_m)\|_1 + \|EM(\cdot)\|_1. \tag{3.43}$$

For abbreviation, we omit $(\boldsymbol{x}, t_m)$. By Eq. (3.16) and the definition of $ER(\boldsymbol{x})$ in (3.40),

$$\begin{aligned} \|\bar{\rho}_1 - \bar{\rho}_1^\delta\|_1 \leq & \ \left\|(\bar{\rho}_1 - \rho_1) - (\bar{\rho}_1^\delta - \rho_1^\delta)\right\|_1 + \|\rho_1 - \rho_1^\delta\|_1 \\ \leq & \ \left\| FTH - \Delta t \sum_{j=1}^{n_f} k_{j,m} L_j \rho_1 + \Delta t k_f \rho_2 - HTF \right\|_1 + E_m \\ \leq & \ \left\| \Delta t \sum_{j=1}^{n_f} k_{j,m} L_j (\rho_1 - \rho_1^\delta) \right\|_1 + \|\Delta t k_f (\rho_2 - \rho_2^\delta)\|_1 + \|ER(\cdot)\|_1 + E_m \\ \leq & \ (1 + \Delta t K) E_m + \|ER(\cdot)\|_1. \end{aligned} \tag{3.44}$$

Now we combine Eq. (3.43) and (3.44) to obtain

$$\|e_1(\cdot, t_{m+1})\|_1 \leq (1 + \Delta t K) E_m + \|ER(\cdot)\|_1 + \|EM(\cdot)\|_1. \tag{3.45}$$

Similarly, by Eq. (3.15) and (3.33),

$$\begin{aligned} \|e_2(\cdot, t_{m+1})\|_1 = & \|\rho_2(\cdot, t_{m+1}) - \rho_2^\delta(\cdot, t_{m+1})\|_1 \\ = & \ \|\bar{\rho}_2(\cdot, t_m) - \bar{\rho}_2^\delta(\cdot, t_m)\|_1 \\ \leq & \ \|(\bar{\rho}_2 - \rho_2) - (\bar{\rho}_2^\delta - \rho_2^\delta)\|_1 + \|\rho_2 - \rho_2^\delta\|_1 \\ \leq & \ (1 + \Delta t K) E_m + \|ER(\cdot)\|_1. \end{aligned} \tag{3.46}$$

Combining Eq. (3.45) and (3.46), we arrive at

$$E_{m+1} \leq (1 + \Delta t K) E_m + \|ER(\cdot)\|_1 + \|EM(\cdot)\|_1.$$

□

In the remainder of this section, we will focus on estimating $\|ER(\cdot)\|_1$ and $\|EM(\cdot)\|_1$.

3.7.1 Error of Motion $\|EM(\cdot)\|_1$

To estimate the L^1 norm, we utilize a spatial discretization. Denote $\bar{\Omega}$ by $[a_0, a_f] \times [b_0, b_f]$, where $b_f - b_0 = a_f - a_0 = \sqrt{|\bar{\Omega}|}$. Select the spatial resolution to be

$$h = \frac{\sqrt{|\bar{\Omega}|}}{\lceil \sqrt{N} \rceil}, \tag{3.47}$$

and discretize $\bar{\Omega}$ into cells of size $h \times h$ as follows:

$$\bar{\Omega}_h = \{(a_0 + ih, b_0 + jh) \in \bar{\Omega} : 0 \leq i, j < \lceil \sqrt{N} \rceil\}. \tag{3.48}$$

We note that there is no spatial discretization in the simulation of the microscopic model, but the selection of h matters in the analysis. The choice of h involves a trade-off: smaller h yields a more accurate estimate, while larger h provides a higher probability that the estimate is true. By Eq. (3.47), there are $\lceil \sqrt{N} \rceil^2$ cells in all, and hence each cell contains one robot on average. For $\boldsymbol{F}(\boldsymbol{x}) : \mathbb{R}^2 \to \mathbb{R}^2 = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}))$, we introduce the infinity norm

$$\|\boldsymbol{F}(\cdot)\|_\infty := \sup\{|f_1(\boldsymbol{x})|, |f_2(\boldsymbol{x})| : \boldsymbol{x} \in \mathbb{R}^2\}.$$

Then we have the following quadrature error.

Lemma 3.7.4 (Quadrature error)**.** *Suppose that $f \in C^\infty(\mathbb{R}^2)$. Then the following inequality holds:*

$$\left| \|f(\cdot)\|_{1,\bar{\Omega}} - \sum_{\boldsymbol{\alpha} \in \bar{\Omega}_h} |f(\boldsymbol{\alpha})| h^2 \right| \;\leq\; 2|\bar{\Omega}| \|\nabla f(\cdot)\|_\infty h.$$

The proof of Lemma 3.7.4 is based on the mean value theorem. A similar estimate can be found in [MB02], Lemma 6.2. In the rest of the section, we always assume that N is sufficiently large for our estimates.

Claim 3.7.5 (Error of motion)**.** *There exists an independent constant, C, such that*

$$\|EM(\cdot)\|_1 \;\leq\; C\Delta t |\bar{\Omega}| \sqrt{D} \delta^{-4} \frac{\ln N}{\sqrt{N}} + \Delta t^2$$

with probability greater than $1 - N^{\left[-\frac{1}{3}(\ln N)\Delta t^2 + 2\right]}$.

Proof of Claim 3.7.5. We note that

$$\|EM(\cdot)\|_1 = \|EM(\cdot)\|_{1,\bar{\Omega}} + \|EM(\cdot)\|_{1,\bar{\Omega}^c}.$$

First, we estimate $\|EM(\cdot)\|_{1,\bar{\Omega}}$. We define the error of motion for the i^{th} robot:

$$Y_i(\boldsymbol{x}) := \frac{1}{N}\int_{\mathbb{R}^2} G_\delta(\boldsymbol{y} - \boldsymbol{X}_m^i)p(\boldsymbol{x}, t_{m+1}|\boldsymbol{y}, t_m)\boldsymbol{dy} - \frac{1}{N}G_\delta(\boldsymbol{x} - \boldsymbol{X}_{m+1}^i) \quad \text{if } i \in F_{m+1}, \quad (3.49)$$

and $Y_i(\boldsymbol{x}) = 0$ if $i \in \cup_j H_{j,m+1}$. Then by Eq. (3.10), (3.14), and the definition of $EM(\boldsymbol{x})$ in Eq. (3.39), we have

$$EM(\boldsymbol{x}) = \sum_{i=1}^{N} Y_i(\boldsymbol{x}).$$

Note that the robots are independent of one another, and thus $Y_i(\boldsymbol{x})$, $i = 1, ..., N$, are independent random variables for any fixed $\boldsymbol{x}$. These random variables have zero mean, i.e.

$$E(Y_i(\boldsymbol{x})) = 0. \quad (3.50)$$

The proof of Eq. (3.50) is given in Subsection 3.7.3, Claim 3.7.9.

Now we apply Bennett's inequality to obtain an upper bound for $|EM(\boldsymbol{x})|$.

Lemma 3.7.6. *(Bennett's inequality) Let Y_i be independent bounded random variables with $E(Y_i) = 0$, $Var(Y_i) = \sigma_i^2$, and $|Y_i| \le M_0$. Let $S = \sum_i Y_i$ and $V \ge \sum_i \sigma_i^2$. Then for $\eta > 0$,*

$$P\{|S| \ge \eta\} \ \le \ 2\exp\left[-\frac{1}{2}\eta^2 V^{-1} B(M_0 \eta V^{-1})\right], \quad (3.51)$$

where $B(\lambda) = 2\lambda^{-2}[(1+\lambda)\ln(1+\lambda) - \lambda]$, $\lambda > 0$, $\lim_{\lambda\to 0^+} B(\lambda) = 1$, and $B(\lambda) \sim 2\lambda^{-1}\ln\lambda$ as $\lambda \to \infty$.

The proof of Lemma 3.7.6 can be found in [Ben62], [Hoe63]. A direct computation yields

$$|Y_i(\boldsymbol{x})| \le \frac{1}{N\delta^2}, \quad \sum_{i=1}^{N} Var(Y_i(\boldsymbol{x})) \le \frac{1}{N\delta^4}.$$

We set $\eta = \frac{\Delta t \ln N}{\sqrt{N}\delta^2}$, $M_0 = N^{-1}\delta^{-2}$, and $V = N^{-1}\delta^{-4}$ in Bennett's inequality to obtain the

following estimate.

$$\begin{aligned} & P\left(|EM(\boldsymbol{x})| \geq \Delta t \frac{\ln N}{\sqrt{N}\delta^2}\right) \\ & \leq 2\exp\left[-\frac{1}{2}\eta^2 V^{-1} B(M_0 \eta V^{-1})\right] \\ & = 2\exp\left[-\frac{1}{2}(\ln N)^2 \Delta t^2 B\left(\Delta t \frac{\ln N}{\sqrt{N}}\right)\right] \\ & \leq 2\exp\left[-\frac{1}{3}(\ln N)^2 \Delta t^2\right] = 2N^{-\frac{1}{3}(\ln N)\Delta t^2}, \end{aligned} \tag{3.52}$$

where we used the fact that $B\left(\Delta t \frac{\ln N}{\sqrt{N}}\right) \geq \frac{2}{3}$ when N is sufficiently large. Hence we have

$$\sum_{\boldsymbol{\alpha}\in\bar{\Omega}_h} |EM(\boldsymbol{\alpha})|h^2 \ \leq \ \Delta t|\bar{\Omega}|\frac{\ln N}{\sqrt{N}\delta^2} \tag{3.53}$$

with probability greater than $1 - 2N^{\left[-\frac{1}{3}(\ln N)\Delta t^2 + 2\right]}$. Next, by Lemma 3.7.4,

$$\left| \|EM(\cdot)\|_{1,\bar{\Omega}} - \sum_{\boldsymbol{\alpha}\in\bar{\Omega}_h} |EM(\boldsymbol{\alpha})|h^2 \right| \leq 2|\bar{\Omega}|\|\nabla EM(\cdot)\|_\infty h \leq 2|\bar{\Omega}| \sum_{i=1}^{N} \|\nabla Y_i(\cdot)\|_\infty h. \tag{3.54}$$

We claim that for each $i = 1, ..., N$,

$$\|\nabla Y_i(\cdot)\|_\infty \leq \ \frac{C_2}{N}\sqrt{D}\Delta t \delta^{-4} \ln N \tag{3.55}$$

with probability greater than $1 - \exp\left[-\frac{1}{2}\Delta t(\ln N)^2\right]$, where C_2 is an independent constant. The proof of this claim is given in Subsection 3.7.3, Claim 3.7.10.

Combining inequalities (3.53), (3.54) and (3.55), and plugging in the choice of h given by Eq. (3.47), we obtain

$$\|EM(\cdot)\|_{1,\bar{\Omega}} \ \leq \ \Delta t \delta^{-2}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + C_2|\bar{\Omega}|^{3/2}\sqrt{D}\Delta t\delta^{-4}\frac{\ln N}{\sqrt{N}} \ \leq \ C\Delta t|\bar{\Omega}|\sqrt{D}\delta^{-4}\frac{\ln N}{\sqrt{N}} \tag{3.56}$$

with probability greater than $1 - 2N^{\left[-\frac{1}{3}(\ln N)\Delta t^2 + 2\right]}$, where C is an independent constant.

Next, we consider $\|EM(\cdot)\|_{1,\bar{\Omega}^c}$. This error is caused by diffusion: in the macroscopic model, the density of *Flying* robots diffuses immediately to the entire $\mathbb{R}^2$, whereas in the microscopic model, the actual density of *Flying* robots always stays in $\bar{\Omega}$.

We claim that:

$$\|EM(\cdot)\|_{1,\bar{\Omega}^c} \ \leq \ \Delta t^2. \tag{3.57}$$

The proof of this claim is given in Subsection 3.7.3, Claim 3.7.11. Now we combine Eq. (3.56) and (3.57) to obtain

$$\|EM(\cdot)\|_1 \;\le\; C\Delta t|\bar{\Omega}|\sqrt{D}\delta^{-4}\frac{\ln N}{\sqrt{N}} + \Delta t^2$$

with probability greater than $1 - N^{\left[-\frac{1}{3}(\ln N)\Delta t^2+2\right]}$. This completes the proof of Claim 3.7.5. □

3.7.2 Error of Reaction $\|ER(\cdot)\|_1$

Claim 3.7.7 (Error of reaction)**.** *There exists an independent constant, C, such that*

$$\|ER(\cdot)\|_1 \;\le\; CK\Delta t\delta^{-3}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} \;+\; \Delta tKP_\delta$$

with probability greater than $1 - CN^{\left[-\frac{1}{3K}(\ln N)\Delta t+2\right]}$.

Proof of Claim 3.7.7. Define an intermediate term

$$FTH'(\boldsymbol{x}, t_m) := \frac{1}{N}\sum_{j=1}^{n_f}\sum_{i\in F_m}\Delta t k_{j,m}L_j(\boldsymbol{X}_m^i)G_\delta(\boldsymbol{x}-\boldsymbol{X}_m^i).$$

Omitting $(\boldsymbol{x}, t_m)$, we have

$$\begin{aligned}|ER(\boldsymbol{x})| &= \left|-\Delta t\sum_{j=1}^{n_f}k_{j,m}L_j(\boldsymbol{x})\rho_1^\delta + \Delta t k_f\rho_2^\delta + FTH - HTF\right| \\ &\le |SE(\boldsymbol{x})| + |OF(\boldsymbol{x})|,\end{aligned}$$

where

$$\begin{aligned}SE(\boldsymbol{x}) &= \left[(FTH - FTH') + (\Delta t k_f\rho_2^\delta - HTF)\right](\boldsymbol{x}, t_m), \\ OF(\boldsymbol{x}) &= \left(\Delta t\sum_{j=1}^{n_f}k_{j,m}L_j(\boldsymbol{x})\rho_1^\delta - FTH'\right)(\boldsymbol{x}, t_m).\end{aligned}$$

Notably, FTH and FTH' are supported in Γ_{out}^δ, while $k_{j,m}L_j(\boldsymbol{x})\rho_1^\delta$ is supported in Γ. Therefore, $OF(\boldsymbol{x})$ measures the *coverage outflow* at the boundary of Γ. FTH and HTF are the actual densities of robot state transitions between *Flying* and *Hovering*, whereas FTH' and

$\Delta t k_f \rho_2^\delta$ are the expected densities of these state transitions; therefore, $SE(\boldsymbol{x})$ measures the *sampling error* of reaction.

First we estimate $\|OF(\cdot)\|_1$. We define $L(\boldsymbol{x}) = \mathbb{1}_\Gamma(\boldsymbol{x})$. Note that by the definitions of $L_j(\boldsymbol{x})$ and $L(\boldsymbol{x})$, we have that

$$L(\boldsymbol{x}) = \sum_{j=1}^{n_f} L_j(\boldsymbol{x}).$$

Now using the definition of ρ_1^δ from Eq. (3.10), we obtain

$$OF(\boldsymbol{x}) = \frac{\Delta t}{N} \sum_{i \in F_m} \sum_{j=1}^{n_f} k_{j,m} (L_j(\boldsymbol{x}) - L_j(\boldsymbol{X}_m^i)) G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i).$$

Let us define

$$Z_i := \begin{cases} \dfrac{1}{N} \displaystyle\sum_{j=1}^{n_f} k_{j,m} \|(L_j(\cdot) - L_j(\boldsymbol{X}_m^i)) G_\delta(\cdot - \boldsymbol{X}_m^i)\|_1 & \text{if } i \in F_m, \\ 0 & \text{if } i \in \cup_{j=1}^{n_f} H_{j,m}, \end{cases}$$

and

$$Z_i' := Z_i - E(Z_i).$$

Then Z_i is coverage outflow of each individual robot, and

$$\|OF(\cdot)\|_1 \leq \ \Delta t \sum_{i=1}^{N} Z_i \ = \ \Delta t \sum_{i=1}^{N} \left(Z_i' + E(Z_i) \right). \tag{3.58}$$

From the fact that $G_\delta(\boldsymbol{x})$ is supported in $\{\boldsymbol{x} : |\boldsymbol{x}| \leq \delta\}$, it is straightforward to see that $(L_j(\boldsymbol{x}) - L_j(\boldsymbol{X}_m^i)) G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i) \equiv 0$ if $\boldsymbol{X}_m^i \notin \Gamma_{out}^\delta - \Gamma_{in}^\delta$. When $\boldsymbol{X}_m^i \in \Gamma_{out}^\delta - \Gamma_{in}^\delta$, we have

$$\begin{aligned} Z_i \ &\leq \ \frac{K}{N} \int_{\mathbb{R}^2} |(L(\boldsymbol{x}) - L(\boldsymbol{X}_m^i)) G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i)| d\boldsymbol{x} \\ &\leq \ \frac{1}{N} \int_{\mathbb{R}^2} G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i) d\boldsymbol{x} \ = \ \frac{K}{N}. \end{aligned}$$

Hence,

$$E(Z_i) \ \leq \ \frac{K}{N} P\left\{ \boldsymbol{X}_m^i \in \Gamma_{out}^\delta - \Gamma_{in}^\delta \right\} \ \leq \ \frac{K}{N} P_\delta. \tag{3.59}$$

Note that Z_i' are i.i.d random variables with $E(Z_i') = 0$, so we can apply Bennett's inequality (Lemma 3.7.6) again to estimate $|\sum_i Z_i'|$. We set $\eta = K \ln N / \sqrt{N}$ and compute M_0 and V

as follows: $|Z_i'| \leq K/N =: M_0$ and $\sum_i Var(Z_i') \leq NM_0^2 = K^2/N =: V$. Plugging η, M_0, and V into Bennett's inequality, we arrive at

$$P\left(\left|\sum_{i=1}^{N} Z_i'\right| \;\geq\; \frac{K\ln N}{\sqrt{N}}\right) \;\leq\; 2\exp\left[-\frac{(\ln N)^2}{3}\right]. \tag{3.60}$$

Combining Eq. (3.58), (3.59), and (3.60), we obtain

$$\|OF(\cdot)\|_1 \;\leq\; \Delta tK\frac{\ln N}{\sqrt{N}} + \Delta tKP_\delta \tag{3.61}$$

with probability greater than $1 - 2\exp\left[-\frac{1}{3}(\ln N)^2\right]$.

Next we estimate $\|SE(\cdot)\|_1$. Since $SE(\boldsymbol{x})$ is supported in $\bar{\Omega}$, we have that $\|SE(\cdot)\|_1 = \|SE(\cdot)\|_{1,\bar{\Omega}}$. We define

$$W_i(\boldsymbol{x}) := \varphi_i G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i), \quad i = 1, ..., N,$$

where

$$\varphi_i := \begin{cases} \dfrac{1}{N}\displaystyle\sum_{j=1}^{n_f}(\Delta tk_{j,m} - I_{i,j,m})L_j(\boldsymbol{X}_m^i) & \text{if } i \in F_m, \\ \dfrac{1}{N}(J_{i,j,m} - \Delta tk_f) & \text{if } i \in H_{j,m}. \end{cases}$$

It can be verified that for a fixed $\boldsymbol{x}$, the random variables $W_i(\boldsymbol{x})$, $i = 1, ..., N$, are independent, and

$$\begin{aligned} &SE(\boldsymbol{x}) = \sum_{i=1}^{N} W_i(\boldsymbol{x}), \quad E(W_i(\boldsymbol{x})) = 0 \\ &|W_i(\boldsymbol{x})| \leq \frac{1}{N\delta^2} \;\text{ and }\; |SE(\boldsymbol{x})| \leq \frac{1}{\delta^2}, \\ &Var(W_i(\boldsymbol{x})) \;\leq\; \delta^{-4}Var(\varphi_i) \;\leq\; \frac{\Delta tK}{N^2\delta^4}. \end{aligned}$$

Using the estimate of quadrature error (Lemma 3.7.4),

$$\begin{aligned} &\left|\|SE(\cdot)\|_{1,\bar{\Omega}} - \sum_{\boldsymbol{\alpha}\in\bar{\Omega}_h} |SE(\boldsymbol{\alpha})|h^2\right| \;\leq\; 2|\bar{\Omega}|\|\nabla SE(\cdot)\|_\infty h \\ &\leq\; 2|\bar{\Omega}|\sum_{i=1}^{N}|\varphi_i|\|\nabla G_\delta(\cdot - \boldsymbol{X}_m^i)\|_\infty h \;\leq\; 5|\bar{\Omega}|\delta^{-3}\sum_{i=1}^{N}|\varphi_i|h. \end{aligned} \tag{3.62}$$

Next we claim that

$$\sum_{i=1}^{N} |\varphi_i| \;\leq\; 2\Delta t K \tag{3.63}$$

with probability greater than $1 - 2\exp\left[-C_1 \Delta t K N/2\right]$, where C_1 is an independent constant. The proof of this claim is given in Subsection 3.7.3, Claim 3.7.12. Now we apply Bennett's inequality again to estimate $SE(\boldsymbol{\alpha})$. Setting $\eta = \Delta t \frac{\ln N}{\sqrt{N}\delta^2}$, $M_0 = \frac{1}{N\delta^2}$, $V = \frac{\Delta t K}{N\delta^4}$ and plugging these parameters into Eq. (3.51), we obtain

$$P\left(|SE(\boldsymbol{\alpha})| \;\geq\; \Delta t \frac{\ln N}{\sqrt{N}\delta^2}\right) \;\leq\; 2N^{\left[-\frac{1}{3K}(\ln N)\Delta t\right]}.$$

Hence,

$$P\left(\sum_{\boldsymbol{\alpha}\in\bar{\Omega}_h} |SE(\boldsymbol{\alpha})|h^2 \;\leq\; \Delta t \delta^{-2} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}}\right) \geq\; 1 - 2N^{\left[-\frac{1}{3K}(\ln N)\Delta t + 2\right]}. \tag{3.64}$$

Combining Eq. (3.62), (3.63), and (3.64), and noting that $\exp(-N) \ll N^{-(\ln N)\Delta t + 2}$, we find that

$$\begin{aligned} \|SE(\cdot)\|_1 \;&\leq\; 10\Delta t \delta^{-3} K \frac{|\bar{\Omega}|^{3/2}}{\sqrt{N}} + \Delta t |\bar{\Omega}| \delta^{-2} \frac{\ln N}{\sqrt{N}} \\ &\leq\; C_2 \Delta t K \delta^{-3} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} \end{aligned} \tag{3.65}$$

with probability greater than $1 - C_3 N^{\left[-\frac{1}{3K}(\ln N)\Delta t + 2\right]}$, where C_2 and C_3 are independent constants. Finally, by combining Eq. (3.61) and (3.65), we conclude that

$$\|ER(\cdot)\|_1 \;\leq\; CK\Delta t \delta^{-3} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} \;+\; \Delta t K P_\delta$$

with probability greater than $1 - CN^{\left[-\frac{1}{3K}(\ln N)\Delta t + 2\right]}$, which completes the proof of Claim 3.7.7. □

We now show how Theorem 3.7.1 follows from Proposition 3.7.3, Claim 3.7.5, and Claim 3.7.7.

Proof of Theorem 3.7.1. Set $\beta = 1 + K\Delta t$. By combining the inequalities in Proposition 3.7.3, Claim 3.7.5, and Claim 3.7.7, we find that

$$E_{m+1} \;\leq\; \beta E_m + C' K \left[\Delta t \delta^{-4} \sqrt{D} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} + \Delta t P_\delta + \Delta t^2\right] \tag{3.66}$$

with probability greater than $1 - C'N^{\left[-\frac{1}{3}(\ln N)\Delta t^2+2\right]}$, where C' is an independent constant. Note that $E_0 = 0$. Iterating over m by using Eq. (3.66), we obtain

$$\begin{aligned} E_{m+1} &\leq C'K\Delta t\frac{\beta^{m+1}-1}{\beta-1}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right] \\ &\leq C'e^{KT}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right] \end{aligned}$$

uniformly in m with probability greater than

$$1 - \frac{C'T}{\Delta t}N^{\left[-\frac{1}{3}(\ln N)\Delta t^2+2\right]}. \tag{3.67}$$

Replacing C' with C, this proves part (i) of the theorem.

To prove part (ii), we start with the following inequality:

$$\begin{aligned} &\|\rho_3(\cdot,t_m) - \rho_3^\delta(\cdot,t_m)\|_1 \\ &= \left\|\sum_{\tau=0}^{m-1}\sum_{j=1}^{n_f} k_{j,\tau}\Delta t L_j(\cdot)\rho_1(\cdot,t_\tau) - \sum_{\tau=0}^{m-1} FTH(\cdot,t_\tau)\right\|_1 \\ &\leq \Delta t\left\|\sum_{\tau=0}^{m-1}\sum_{j=1}^{n_f} k_{j,\tau}L_j(\cdot)(\rho_1(\cdot,t_\tau) - \rho_1^\delta(\cdot,t_\tau))\right\|_1 \\ &\qquad + \left\|\sum_{\tau=0}^{m-1}\sum_{j=1}^{n_f} \Delta t k_{j,\tau}L_j(\cdot)\rho_1^\delta(\cdot,t_\tau) - \sum_{\tau=0}^{m-1} FTH(\cdot,t_\tau)\right\|_1 \\ &:= \Lambda_1 + \Lambda_2. \end{aligned}$$

Here, Λ_1 is the cumulative error in the positions of the *Flying* robots, and Λ_2 is the cumulative error in reactions. We have that

$$\begin{aligned} \Lambda_1 &\leq \Delta t K\sum_{\tau=0}^{m-1} E_\tau \\ &\leq \Delta t K\sum_{\tau=0}^{m-1} C'e^{KT}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right] \\ &= C'TKe^{KT}\left[\delta^{-4}\sqrt{D}|\bar{\Omega}|\frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t\right]. \end{aligned} \tag{3.68}$$

Next we estimate Λ_2. We have

$$\Lambda_2 \;\leq\; \sum_{\tau=0}^{m-1}\left\|\sum_{i=1}^{n_f}\Delta t k_{j,\tau}L_j(\cdot)\rho_1^\delta - FTH(\cdot,t_\tau)\right\|_1.$$

We note that the term $\sum_{i=1}^{n_f} \Delta t k_{j,\tau} L_j(\boldsymbol{x}) \rho_1^\delta - FTH(\boldsymbol{x}, t_\tau)$ comprises part of the error of reaction $ER(\boldsymbol{x})$, according to Eq. (3.40). Using an argument similar to the one in Claim 3.7.7, we obtain

$$\begin{aligned} \Lambda_2 &\leq \sum_{\tau=0}^{m-1} \left[CK\Delta t \delta^{-3} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} + \Delta t K P_\delta \right] \\ &= T \left[CK\delta^{-3} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} + K P_\delta \right]. \end{aligned} \tag{3.69}$$

Combining Eq. (3.68) and (3.69), we arrive at our conclusion:

$$\|e_3(\cdot, t_m)\|_1 \leq C'' K T e^{KT} \left[\delta^{-4} \sqrt{D} |\bar{\Omega}| \frac{\ln N}{\sqrt{N}} + P_\delta + \Delta t \right]$$

uniformly in m with probability greater than expression (3.67). Replacing C'' with C, this completes the proof of Theorem 3.7.1. □

Thus far, we have presented an estimate of the L^1 error between the expected density field ρ_3 and the actual density field ρ_3^δ. We can compute the relative error between these density fields as:

$$REL = \frac{\|\rho_3(\cdot, T) - \rho_3^\delta(\cdot, T)\|_1}{\|\rho_3(\cdot, T)\|_1}. \tag{3.70}$$

In practice, however, we would want to compare ρ_3^δ to the *target distribution* ρ_Ω. Moreover, since the user will be satisfied as long as the crops are sufficiently pollinated, we can consider the over-pollinated portion as an inefficiency rather than an error. Hence, we only count insufficient pollination as error. We define the *discrepancy* γ and *efficiency* as

$$\gamma = \frac{\|\rho_3^\delta(\cdot, T) \wedge \rho_\Omega(\cdot) - \rho_\Omega(\cdot)\|_1}{\|\rho_\Omega(\cdot)\|_1}, \tag{3.71}$$

$$\text{Efficiency} = \frac{\|\rho_3^\delta(\cdot, T) \wedge \rho_\Omega(\cdot)\|_1}{\|\rho_3^\delta(\cdot, T)\|_1}, \tag{3.72}$$

where $\rho_3^\delta(\boldsymbol{x}, T) \wedge \rho_\Omega(\boldsymbol{x}) = \min\{\rho_3^\delta(\boldsymbol{x}, T), \rho_\Omega(\boldsymbol{x})\}$. We also define the *intrinsic discrepancy*, which does not depend on N and δ, as

$$\gamma_\Omega = \frac{\|\rho_3(\cdot, T) \wedge \rho_\Omega(\cdot) - \rho_\Omega(\cdot)\|_1}{\|\rho_\Omega(\cdot)\|_1}. \tag{3.73}$$

Note that we do not analyze the error between the target distribution and the macroscopic model in this chapter, since the optimal control is already studied in [EB15]. Given a *desired discrepancy* γ_d, our goal is to select N and δ such that

$$\gamma \ \leq \ \gamma_d. \tag{3.74}$$

Toward this end, we present the following corollary.

Corollary 3.7.8. *Under the same assumptions as in Theorem 3.7.1, we have that*

$$\gamma \ \leq \ \gamma_\Omega + C_4 \left(\delta + \Delta t + (1 + \delta^{-4}) \frac{\ln N}{\sqrt{N}} \right) \tag{3.75}$$

with probability greater than $1 - \frac{CT}{\Delta t} N^{\left[-\frac{1}{3}(\ln N)\Delta t^2 + 2\right]}$. *Here,* C_4 *is a constant that depends on* $k_j(t)$, D, Γ_j, T, *and* $\bar{\Omega}$.

Proof. Since $|\min\{a, c\} - \min\{b, c\}| \leq |a - b|$,

$$\begin{aligned}
&\|\rho_3^\delta \wedge \rho_\Omega(\cdot, T) - \rho_\Omega(\cdot)\|_1 \\
\leq &\|\rho_3 \wedge \rho_\Omega(\cdot, T) - \rho_\Omega(\cdot)\|_1 + \|\rho_3^\delta \wedge \rho_\Omega(\cdot, T) - \rho_3 \wedge \rho_\Omega(\cdot, T)\|_1 \\
\leq &\|\rho_3 \wedge \rho_\Omega(\cdot, T) - \rho_\Omega(\cdot)\|_1 + \|\rho_3^\delta(\cdot, T) - \rho_3(\cdot, T)\|_1.
\end{aligned}$$

Hence,

$$\gamma \ \leq \ \gamma_\Omega + \frac{\|\rho_3^\delta(\cdot, T) - \rho_3(\cdot, T)\|_1}{\|\rho_\Omega(\cdot)\|_1}.$$

By Theorem 3.7.1, $\exists$ an independent constant $C_4' > 0$ such that

$$\|\rho_3^\delta(\cdot, T) - \rho_3(\cdot, T)\|_1 \ \leq \ C_4' \left(\delta + \Delta t + (1 + \delta^{-4}) \frac{\ln N}{\sqrt{N}} \right).$$

Divide both sides by $\|\rho_\Omega(\cdot)\|_1$ and apply it to the previous inequality, and Eq. (3.75) is proved. □

3.7.3 Proof of the Claims

Claim 3.7.9. *Let* $Y_i(\boldsymbol{x})$ *be defined as in Eq.* (3.49). *Then for each* $i = 1, ..., N$ *and each* $\boldsymbol{x}$,

$$E(Y_i(\boldsymbol{x})) = 0.$$

Proof. Note that if $i \in F_{m+1}$,

$$\begin{aligned}
&E\left(G_\delta(\boldsymbol{x}-\boldsymbol{X}_{m+1}^i)\right)\\
=&E\left\{E\left[G_\delta\left(\boldsymbol{x}-\boldsymbol{X}_m^i-\boldsymbol{v}(t_m)\Delta t-\sqrt{2D\Delta t}\Delta\boldsymbol{Z}_m\right)\middle|\boldsymbol{X}_m^i\right]\right\}\\
=&E\left[\int_{\mathbb{R}^2}G_\delta\left(\boldsymbol{x}-\boldsymbol{X}_m^i-\boldsymbol{v}(t_m)\Delta t-\boldsymbol{y}\right)\frac{1}{4\pi D\Delta t}e^{-\frac{|\boldsymbol{y}|^2}{4D\Delta t}}\boldsymbol{dy}\right]\\
=&E\left[\int_{\mathbb{R}^2}G_\delta(\boldsymbol{y}'-\boldsymbol{X}_m^i)\frac{1}{4\pi D\Delta t}e^{-\frac{|\boldsymbol{x}-\boldsymbol{v}(t_m)\Delta t-\boldsymbol{y}'|^2}{4D\Delta t}}\boldsymbol{dy}'\right]\\
=&E\left[\int_{\mathbb{R}^2}G_\delta(\boldsymbol{y}-\boldsymbol{X}_m^i)p(\boldsymbol{x},t_{m+1}|\boldsymbol{y},t_m)\boldsymbol{dy}\right],
\end{aligned}$$

where we applied the change of variable $\boldsymbol{y}'=\boldsymbol{x}-\boldsymbol{v}(t_m)-\boldsymbol{y}$. This proves our statement for $i \in F_{m+1}$. In addition, $Y_i(\boldsymbol{x})=0$ for each $i \notin F_{m+1}$. □

Claim 3.7.10. *For each $i=1,...,N$,*

$$\|\nabla Y_i(\cdot)\|_\infty \leq \frac{C}{N}\sqrt{D}\Delta t\delta^{-4}\ln N$$

with probability greater than $1-\exp\left[-\frac{1}{2}\Delta t(\ln N)^2\right]$, where C is an independent constant.

Proof. It is straightforward to see that for each $i \in \cup_j H_{j,m+1}$, $\nabla Y_i(\boldsymbol{x})=0$. For each $i \in F_{m+1}$,

$$\nabla Y_i(\boldsymbol{x})=\frac{1}{N}\int_{\mathbb{R}^2}\nabla G_\delta(\boldsymbol{x}'-\boldsymbol{y})p(\boldsymbol{y};\Delta t)\boldsymbol{dy}-\frac{1}{N}\nabla G_\delta(\boldsymbol{x}'-\sqrt{2D\Delta t}\Delta\boldsymbol{Z}_m),$$

where

$$\begin{aligned}
\boldsymbol{x}'&=\boldsymbol{x}-\boldsymbol{v}(t_m)\Delta t-\boldsymbol{X}_m^i,\\
p(\boldsymbol{y};\Delta t)&=\frac{1}{4\pi D\Delta t}\exp\left(-\frac{|\boldsymbol{y}|^2}{4D\Delta t}\right).
\end{aligned}$$

Note that

$$|\nabla Y_i(\boldsymbol{x})| \leq W_1+W_2,$$

with

$$\begin{aligned}
W_1&=\frac{1}{N}\int_{\mathbb{R}^2}|\nabla G_\delta(\boldsymbol{x}'-\boldsymbol{y})-\nabla G_\delta(\boldsymbol{x}'-\sqrt{2D\Delta t}\Delta\boldsymbol{Z}_m-\boldsymbol{y})|p(\boldsymbol{y};\Delta t)\boldsymbol{dy}\\
W_2&=\frac{1}{N}\left|\int_{\mathbb{R}^2}\nabla G_\delta(\boldsymbol{x}'-\sqrt{2D\Delta t}\Delta\boldsymbol{Z}_m-\boldsymbol{y})p(\boldsymbol{y};\Delta t)\boldsymbol{dy}-\nabla G_\delta(\boldsymbol{x}'-\sqrt{2D\Delta t}\Delta\boldsymbol{Z}_m)\right|.
\end{aligned}$$

Since $\Delta \boldsymbol{Z}_m = (\Delta Z_m^1, \Delta Z_m^2) \sim \boldsymbol{W}(0,1)$,

$$P(\max_{i=1,2} |\Delta Z_m^i| \le \sqrt{\Delta t} \ln N) \ \ge \ 1 - \exp\left[-\frac{1}{2}\Delta t (\ln N)^2\right].$$

By the mean value theorem,

$$\begin{aligned} W_1 &= \frac{1}{N}\int_{\mathbb{R}^2} \left|\partial^2 G_\delta(\xi_1(\boldsymbol{x},\boldsymbol{y}))\sqrt{2D\Delta t}\Delta \boldsymbol{Z}_m p(\boldsymbol{y};\Delta t)\right| \boldsymbol{dy} \\ &\le \ \frac{2\sqrt{2D\Delta t}}{N} \max_{i,j=1,2} |\partial_{x_i x_j} G_\delta| \max_{i=1,2} |\Delta Z_m^i| \\ &\le \ \frac{C'}{N}\sqrt{D}\Delta t \delta^{-4} \ln N \end{aligned} \tag{3.76}$$

with probability greater than $1 - \exp\left[-\frac{1}{2}\Delta t (\ln N)^2\right]$. Here C' is an independent constant.

Now we estimate W_2. It is straightforward to see that

$$W_2 \ \le \ \|\nabla G_\delta * p - \nabla G_\delta\|_\infty.$$

By using a change of variable $\boldsymbol{y} = \sqrt{\Delta t}\boldsymbol{y}'$, we obtain

$$\begin{aligned} & G_\delta * p(\boldsymbol{x};\Delta t) - G_\delta(\boldsymbol{x}) \\ &= \int_{\mathbb{R}^2} \left[-\partial^2 G_\delta(\boldsymbol{x})\sqrt{\Delta t}\boldsymbol{y}' + \frac{1}{2}\Delta t \partial^3 G_\delta(\xi_2(\boldsymbol{x},\boldsymbol{y}'))|\boldsymbol{y}'|^2\right] \cdot p(\boldsymbol{y}';1)\boldsymbol{dy}'. \end{aligned}$$

Applying the facts that

$$\int_{\mathbb{R}^2} \boldsymbol{y}' p(\boldsymbol{y}';1)\boldsymbol{dy}' \ = \ 0, \quad \max_{i,j,z=1,2} |\partial_{x_i x_j x_z} G_\delta| \ \le \ C\delta^{-5},$$

we have that

$$W_2 \ \le \ \|\nabla G_\delta * p - \nabla G_\delta\|_\infty \ \le \ C''\Delta t \delta^{-5}, \tag{3.77}$$

where C'' is an independent constant. Combining Eq. (3.76) and (3.77), and noting that $N \gg \delta^{-1}$, we arrive at

$$\|\nabla Y_i(\cdot)\|_\infty \ \le \ \frac{C}{N}\sqrt{D}\Delta t \delta^{-4} \ln N$$

with probability greater than $1 - \exp\left[-\frac{1}{2}\Delta t (\ln N)^2\right]$. □

Claim 3.7.11. *The following inequality holds:*

$$\|EM(\cdot)\|_{1,\bar{\Omega}^c} \;\leq\; \Delta t^2.$$

Proof. By our assumptions in Theorem 3.7.1, $\exists \zeta > 0$ such that $\boldsymbol{X}_m^i \in \bar{\Omega}_{in}^{2\zeta}$ and $\delta < \zeta$. Since $G_\delta(\boldsymbol{x})$ is supported in B_δ,

$$G_\delta(\boldsymbol{x} - \boldsymbol{X}_m^i) = 0 \quad \forall \boldsymbol{x} \notin \bar{\Omega}_{in}^{\zeta},\ \forall m = 1, ..., M,\ i = 1, ..., N.$$

Therefore, we have

$$\|Y_i(\cdot)\|_{1,\bar{\Omega}^c} = \frac{1}{N} \int_{\bar{\Omega}^c} \int_{\bar{\Omega}_{in}^{\zeta}} G_\delta(\boldsymbol{y} - \boldsymbol{X}_m^i) p(\boldsymbol{x}, t_{m+1} | \boldsymbol{y}, t_m) \cdot \boldsymbol{dy dx}.$$

By definition, $\forall \boldsymbol{x} \in \bar{\Omega}^c, \boldsymbol{y} \in \bar{\Omega}_{in}^{\zeta}$, $|\boldsymbol{x} - \boldsymbol{y}| \geq \zeta$. Thus, we can choose Δt small enough so that $|\boldsymbol{x} - \boldsymbol{y} - v(t_m)\Delta t| \geq \frac{\zeta}{2}$. Defining $B_\zeta = \{\boldsymbol{x} : |\boldsymbol{x}| \leq \zeta\}$, we have

$$\begin{aligned}
\|Y_i(\cdot)\|_{1,\bar{\Omega}^c} &\leq \frac{1}{N} \int_{\bar{\Omega}_{in}^{\zeta}} G_\delta(\boldsymbol{y} - \boldsymbol{X}_m^i) \boldsymbol{dy} \int_{(B_{\zeta/2})^c} \frac{1}{4\pi D \Delta t} e^{-\frac{|\boldsymbol{x}|^2}{4D\Delta t}} \boldsymbol{dx} \\
&\leq \frac{1}{N} \int_{\zeta/2}^{\infty} \frac{r}{2D\Delta t} e^{-\frac{r^2}{4D\Delta t}} \boldsymbol{dr} \;=\; \frac{1}{N} e^{-\frac{\zeta^2}{16D\Delta t}} \;\leq\; \frac{\Delta t^2}{N}
\end{aligned}$$

since $\Delta t \ll \zeta$. Hence,

$$\|EM(\cdot)\|_{1,\bar{\Omega}^c} \;\leq\; \sum_{i=1}^{N} \|Y_i(\cdot)\|_{1,\bar{\Omega}^c} \;\leq\; \Delta t^2.$$

□

Claim 3.7.12. *The inequality*

$$\sum_{i=1}^{N} |\varphi_i| \;\leq\; 2\Delta t K$$

is true with probability greater than $1 - 2\exp[-C\Delta t K N/2]$*, where* C *is an independent constant.*

Proof. We define

$$\varphi_i' = |\varphi_i| - E(|\varphi_i|).$$

Then $E(\varphi_i') = 0$ and $E(|\varphi_i|) \leq \Delta t K N^{-1}$. Applying Bennett's inequality (3.51) with $\eta = \Delta t K$, $M_0 = N^{-1}$, and $V = \Delta t K N^{-1}$, we obtain

$$P\left(\left|\sum_{i=1}^{N} \varphi_i'\right| \geq \Delta t K\right) \;\leq\; 2\exp\left[-B(1)\Delta t K N/2\right].$$

Hence,

$$\sum_{i=1}^{N} |\varphi_i| \;\leq\; \left|\sum_{i=1}^{N} \varphi_i'\right| + \sum_{i=1}^{N} E(|\varphi_i|) \;\leq\; 2\Delta t K$$

with probability greater than $1 - 2\exp\left[-B(1)\Delta t K N/2\right]$. □

3.8 Simulation Results

In this section, we illustrate the design procedure in Section 3.3 for a simulated crop pollination scenario. Simulation results beyond those required for the design procedure are also presented to validate our convergence analysis.

1) *Set the parameter values.*

We set the example crop field to be a unit square, $\Omega = [0,1]^2$, which has five rows of $n_f = 2$ different types of crops. The regions of type 1 crops and type 2 crops are defined, respectively, as $\Gamma_1 = \{(x_1, x_2) : x_1 \in [0.05, 0.15] \cup [0.45, 0.55] \cup [0.85, 0.95], x_2 \in [0.05, 0.95]\}$ and $\Gamma_2 = \{(x_1, x_2) : x_1 \in [0.25, 0.35] \cup [0.65, 0.75], x_2 \in [0.05, 0.95]\}$. Let the target pollination distribution be

$$\rho_\Omega(\boldsymbol{x}) = 6 \cdot \mathbb{1}_{\Gamma_1}(\boldsymbol{x}) + 12 \cdot \mathbb{1}_{\Gamma_2}(\boldsymbol{x}), \tag{3.78}$$

which is shown in the top left of Fig. 3.4. The other simulation parameters are $\boldsymbol{X}_0 = (0.4, 0.2)$, $T = 240$, $k_f = 0.2$, $D = 0.0005$, $\gamma_d = 0.25$, $v_1^{min} = v_2^{min} = -0.01$, $v_1^{max} = v_2^{max} = 0.01$, and $\Delta t = 0.5$.

We note that our choice of $\Delta t = 0.5$ is based on empirical tests of a range of Δt values, each of which satisfies the CFL condition needed to solve the advection operator in the macroscopic PDE model. We found that the numerical solution of the macroscopic model does not change significantly for $\Delta t \in (0, 1.5]$. This is because, as shown in Fig. 3.3, the robots' optimized velocity components $v_1(t), v_2(t)$ and pollination rates $k_1(t), k_2(t)$ do not

display sharp variations over any time period of 1.5 units, which means that the typical time scales of $v_1(t), v_2(t), k_1(t)$, and $k_2(t)$ are much larger than the time scale resolved with a choice of $\Delta t \in (0, 1.5]$. Furthermore, we note that the choice of Δt has little effect on the error bounds (3.37) and (3.38), as explained in Remark 3.7.2.

2) *Apply the optimal control technique to compute the robot control policies.*

Using the parameters above, we run the optimal control technique described in Section 3.6 to compute the robots' velocity $\boldsymbol{v}(t)$ and pollination rates $k_1(t)$ and $k_2(t)$, which are plotted in Fig. 3.3.

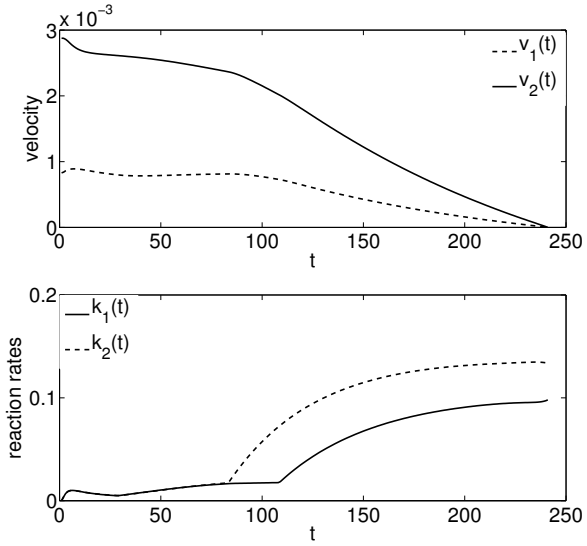


Figure 3.3: *Top*: robot velocity field $\boldsymbol{v}(t) = (v_1(t), v_2(t))$. *Bottom*: robot pollination rates $k_1(t)$ and $k_2(t)$. [ZBE18] Copyright ©2017 IEEE.

3) *Simulate the microscopic model.*

We simulate the microscopic model with the optimized values of $\boldsymbol{v}(t)$, $k_1(t)$, and $k_2(t)$ from Step 2 and the robot sensing radius $\delta = 0.015$. While the design procedure only requires simulations for two distinct values of the swarm size N, here we simulate the microscopic model for all the values of N shown in Table 3.1. We run 100 simulation trials for each value of N.

4) *Compute the discrepancy γ between the actual and target pollination distributions.*

For each simulation of the microscopic model, we compute the resulting actual pollination density field, $\rho_3^\delta(\boldsymbol{x}, T)$, and calculate the discrepancy γ from Eq. (3.71), the relative error REL from Eq. (3.70), and the efficiency from Eq. (3.72). Table 3.1 shows the mean γ, REL, and efficiency for each value of N over 100 simulation trials.

Note that as the swarm size N increases, the mean values of γ and REL decrease. This is due to the convergence of the actual pollination density $\rho_3^\delta(\boldsymbol{x}, T)$ to the expected pollination density $\rho_3(\boldsymbol{x}, T)$ with increasing N. We illustrate this convergence in Fig. 3.4, which plots $\rho_3^\delta(\boldsymbol{x}, T)$ resulting from several values of N (one simulation trial per N) alongside $\rho_3(\boldsymbol{x}, T)$ and the target distribution $\rho_\Omega(\boldsymbol{x})$. To obtain $\rho_3(\boldsymbol{x}, T)$, we numerically solved the macroscopic model over the domain $\bar{\Omega} = [-1, 2]^2$ with $h = 0.006$. The intrinsic discrepancy for this scenario was computed to be $\gamma_\Omega = 0.1413$.

Table 3.1: Simulation results with respect to different N.

N	Mean γ	Mean REL	Mean Efficiency
6400	0.1912	0.1968	0.8881
3200	0.2139	0.2636	0.8667
1600	0.2486	0.3601	0.8257
800	0.3065	0.4997	0.7615
400	0.3924	0.7017	0.6643
200	0.5284	0.9845	0.5177

5) *Estimate the required N such that the discrepancy γ is less than γ_d.*

From Corollary 3.7.8, we have that

$$\gamma \;\leq\; c_1' + c_2' \frac{\ln N}{\sqrt{N}}, \tag{3.79}$$

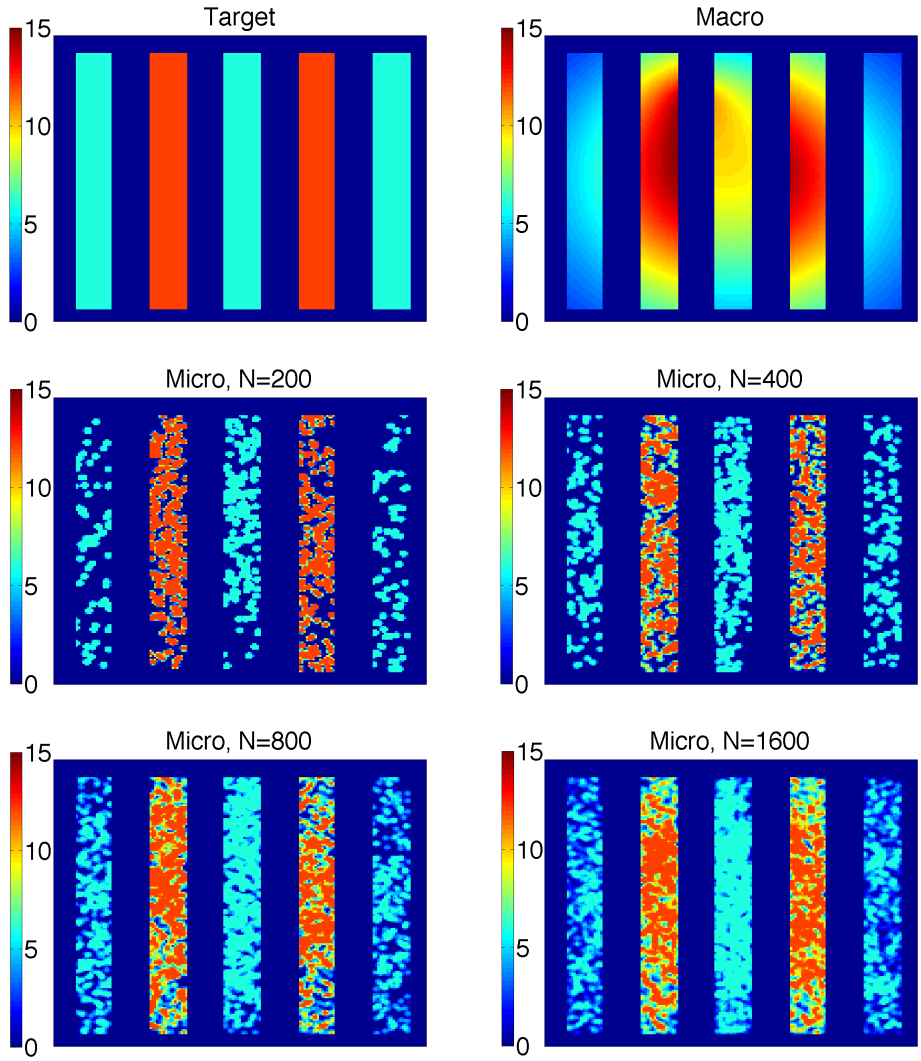


Figure 3.4: *Top left*: Target pollination distribution $\rho_\Omega(\boldsymbol{x})$. *Top right*: Expected pollination distribution $\rho_3(\boldsymbol{x}, T)$ from the macroscopic model. *From middle left to bottom right*: Actual pollination distribution $\rho_3^\delta(\boldsymbol{x}, T)$ from the microscopic model with $\delta = 0.015$ and $N = 200$, 400, 800, 1600 robots. The field is $[0, 1]^2$. [ZBE18] Copyright ©2017 IEEE.

where c_1' is the error determined by Δt and δ, and c_2' is a coefficient that depends on δ. Since

$$\ln N \ll N$$

when N is sufficiently large, we conjecture that

$$\gamma \;=\; c_1 + c_2 \frac{1}{\sqrt{N}}. \tag{3.80}$$

In the bottom subfigure of Fig. 3.5, the linear fitting of mean γ against $1/\sqrt{N}$ verifies Eq. (3.80). This figure also shows that there is a linear relationship between the mean value of REL and $1/\sqrt{N}$.

Now for $\gamma_d = 0.25$, we show how to select the number of robots that are needed to achieve the specification (3.74). We solve for c_1, c_2 in Eq. (3.80) using the mean γ for $N_1 = 200$, $N_2 = 400$ from Table 3.1. The resulting two equations,

$$0.5284 = c_1 + c_2 \frac{1}{\sqrt{N_1}}, \quad 0.3924 = c_1 + c_2 \frac{1}{\sqrt{N_2}}, \tag{3.81}$$

yield $c_1 = 0.06407$, $c_2 = 6.567$. We plug these coefficients into Eq. (3.80) and choose the smallest N such that $\gamma \;\leq\; \gamma_d = 0.25$. This yields $N \approx 1249$.

Remark 3.8.1. *The robots in this scenario act independently of one another, since there are no interactions such as communication. Hence, a swarm with a large population N will achieve the same distribution of pollination over one deployment as a swarm with a smaller population of αN, $\alpha \in (0, 1)$, over α^{-1} deployments. This deployment strategy can be used when the required value of N for some γ_d exceeds the number of available robots.*

6) *Select the value of δ that yields the minimum γ for the required N.*

For the selected value of N, there exists an optimal value of δ that yields a minimum value of the discrepancy γ for that N. We illustrate this in Fig. 3.6, which plots the mean value of γ over 100 simulation trials with respect to different pairs of δ and N. We note that the range of δ in our study and the choice of $\Delta t = 0.5$ yield a very small error (< 0.01) in the operating splitting method (3.29).

Fig. 3.6 reflects a trade-off in choosing δ that is predicted by our error analysis: for a given swarm size N, small δ yield a low coverage outflow near the crop boundary but a high coverage insufficiency, whereas large δ yield a high coverage outflow and a low coverage insufficiency. As the plot shows, the optimal δ becomes smaller as the swarm size N increases. From Step 5, we find that $N \approx 1249$ is the smallest N for which the discrepancy does not exceed γ_d. Thus, we can choose any $N > 1249$, such as $N = 1600$. Then, from Fig. 3.6, we can pick the optimal δ for $N = 1600$ to further decrease the discrepancy, which gives us $\delta \approx 0.024$ and $\gamma \approx 0.23$. In practice, the sensor limitations will impose an upper bound on δ that may be lower than the optimal value. For example, if we choose $N = 1600$ and the possible range of δ for the sensor is $[0, 0.020]$, then according to Fig. 3.6, we should choose δ to be 0.020 instead of 0.024.

We further illustrate the effect of δ with the results in Fig. 3.7, which plots $\rho_3^\delta(\boldsymbol{x}, T)$ resulting from a relatively small robot population $N = 100$ and several values of δ (one simulation trial per δ) alongside the target distribution $\rho_\Omega(\boldsymbol{x})$. The figure shows that when N is fixed at 100, the discrepancy γ is very large and coverage is fairly sparse when $\delta = 0.015$, and increasing δ to 0.030 yields a lower discrepancy and improved coverage.

3.9 Conclusion

In this chapter, we derived analytical bounds on the error between a target spatial distribution of coverage activity and the actual coverage distribution that is achieved by a swarm of N robots whose population dynamics can be described by an Advection-Diffusion-Reaction PDE. We consider scenarios in which the environment is known and the robots' capabilities are highly constrained, in that they have no inter-robot communication or global position information. The analytical bound revealed an almost linear relationship between the coverage error and $N^{-\frac{1}{2}}$, thus providing a convenient way to choose a swarm size that produces a coverage distribution within a maximum allowable error. Our analysis also indicated the existence of an optimal robot sensing radius that minimizes the discrepancy between the actual and target coverage distributions for each swarm size, which provides a theoretical

basis for selecting a particular sensing range. We verified our analytical results through simulations of a crop pollination scenario. We hope that the detailed analysis presented here will inspire the analysis and design of other distributed systems with a significant stochastic component.

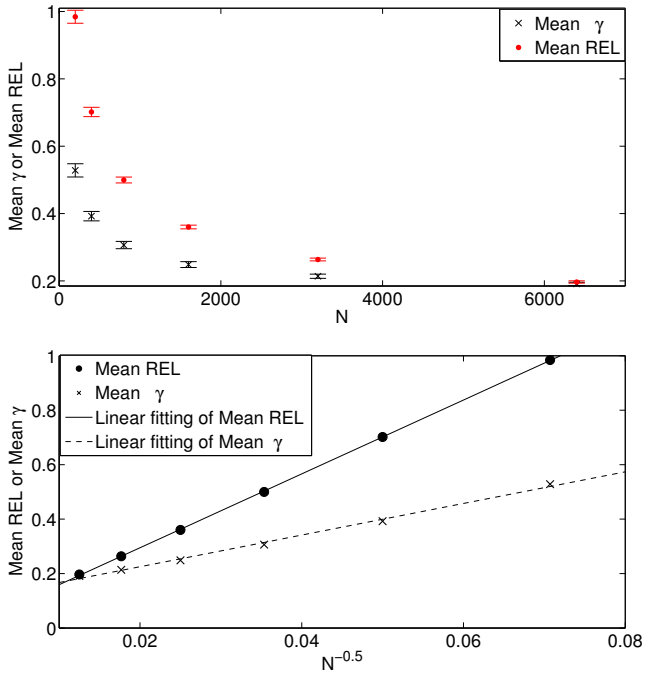


Figure 3.5: *Top figure*: Swarm size N vs. mean REL and mean γ, both averaged over 100 simulations of the microscopic model for each value of N. The corresponding standard deviations are shown as error bars. *Bottom figure*: $1/\sqrt{N}$ vs. mean REL and mean γ. The solid and dashed lines are the linear fittings of mean REL and mean γ, respectively. [ZBE18]

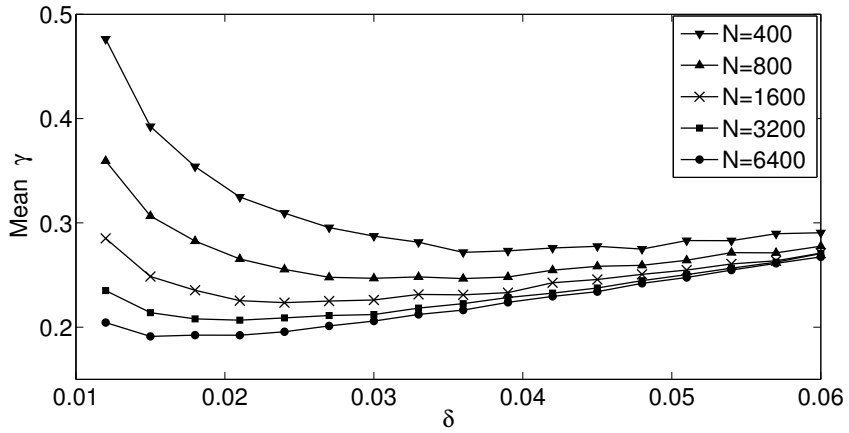


Figure 3.6: Relationship among δ, N, and γ. Each data point is averaged over 100 simulations of the microscopic model with the corresponding values of N and δ. [ZBE18] Copyright ©2017 IEEE.

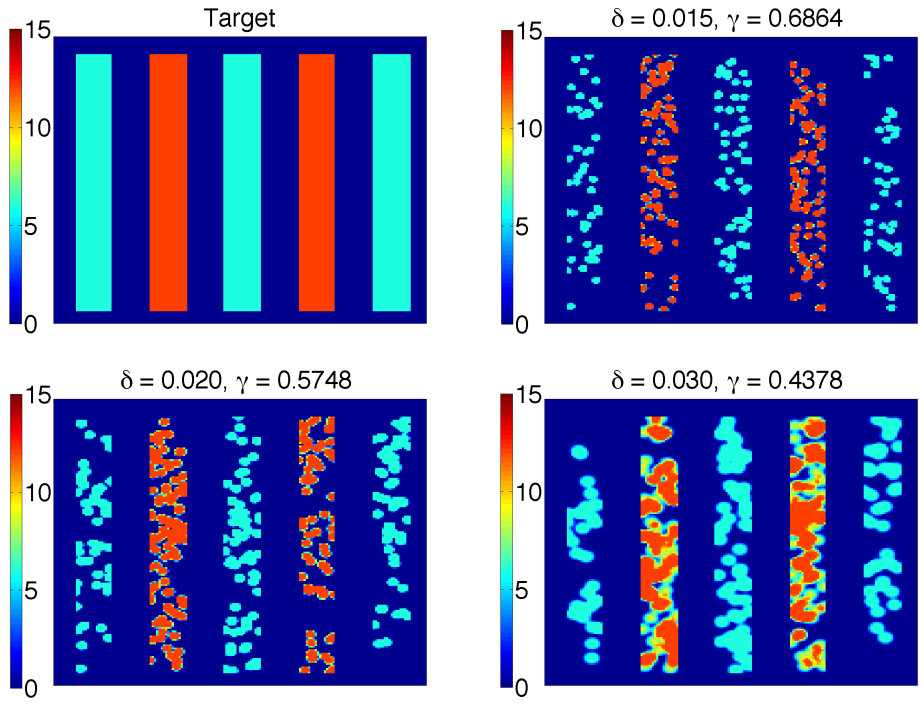


Figure 3.7: *Top left*: Target pollination distribution $\rho_\Omega(\boldsymbol{x})$. *From top right to bottom right*: Actual pollination distribution $\rho_3^\delta(\boldsymbol{x}, T)$ from the microscopic model with $N = 100$ robots and $\delta = 0.015$, 0.020, 0.030. The discrepancy γ is shown for each value of δ. The field is $[0, 1]^2$. [ZBE18] Copyright ©2017 IEEE.

REFERENCES

[AB12] Behçet Açikmeşe and David S Bayard. "A markov chain approach to probabilistic swarm guidance." In *American Control Conference (ACC), 2012*, pp. 6300–6307. IEEE, 2012.

[AB13] M. Annunziato and A. Borzì. "A Fokker–Planck control framework for multidimensional stochastic processes." *Journal of Computational and Applied Mathematics*, **237**(1):487–507, 2013.

[APB14] Pushkarini Agharkar, Rushabh Patel, and Francesco Bullo. "Robotic surveillance and Markov chains with minimal first passage time." In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 6603–6608. IEEE, 2014.

[BCH13] Saptarshi Bandyopadhyay, Soon-Jo Chung, and Fred Y Hadaegh. "Inhomogeneous Markov chain approach to probabilistic swarm guidance algorithm." In *5th Int. Conf. Spacecraft Formation Flying Missions and Technologies*, 2013.

[BDT99] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. SFI Studies in the Science of Complexity. Oxford University Press, Inc., New York, NY, 1999.

[Ben62] G. Bennett. "Probability inequalities for the sum of independent random variables." *Journal of the American Statistical Association*, **57**(297):33–45, 1962.

[BFB13] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo. "Swarm robotics: a review from the swarm engineering perspective." *Swarm Intelligence*, pp. 1–41, 2013.

[BFY13] A. Bensoussan, J. Frehse, and P. Yam. *Mean Field Games and Mean Field Type Control Theory*, volume 101. Springer-Verlag New York, 2013.

[BK04] Calin Belta and Vijay Kumar. "Abstraction and control for groups of robots." *IEEE Transactions on robotics*, **20**(5):865–875, 2004.

[BKN11] Spring Berman, Vijay Kumar, and Radhika Nagpal. "Design of control policies for spatially inhomogeneous robot swarms with application to commercial pollination." In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 378–385, 2011.

[BNH11] Spring Berman, Radhika Nagpal, and Adám Halász. "Optimization of stochastic strategies for spatially inhomogeneous robot swarms: a case study in commercial pollination." In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3923–3930, 2011.

[BV15] J. Beal and M. Viroli. "Space–time programming." *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, **373**(2046), 2015.

[CB16] K. Craig and A. L. Bertozzi. "A blob method for the aggregation equation." *Mathematics of Computation*, **85**(300):1681–1717, 2016.

[CCD16] T. H. Chung, M. R. Clement, M. A. Day, K. D. Jones, D. Davis, and M. Jones. "Live-fly, large-scale field experimentation for large numbers of fixed-wing UAVs." In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1255–1262, May 2016.

[CFT08] C. Canuto, F. Fagnani, and P. Tilli. "A Eulerian approach to the analysis of rendez-vous algorithms." In *Proceedings of the 17th IFAC World Congress*, pp. 9039–9044, 2008.

[CJL06] XZ Cheng, MBA Jalil, Hwee Kuan Lee, and Yutaka Okabe. "Mapping the Monte Carlo scheme to Langevin dynamics: a Fokker-Planck approach." *Physical review letters*, **96**(6):067208, 2006.

[CMK04] Jorge Cortes, Sonia Martinez, Timur Karatas, and Francesco Bullo. "Coverage control for mobile sensing networks." *IEEE Transactions on robotics and Automation*, **20**(2):243–255, 2004.

[CP12] R. M. Colombo and N. Pogodaev. "Confinement strategies in a model for the interaction between individuals and a continuum." *SIAM Journal on Applied Dynamical Systems*, **11**(2):741–770, 2012.

[DBK12] Karthik Dantu, Spring Berman, Bryan Kate, and Radhika Nagpal. "A comparison of deterministic and stochastic approaches for allocating spatially dependent tasks in micro-aerial vehicle collectives." In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 793–800, 2012.

[EAB16] K. Elamvazhuthi, C. Adams, and S. Berman. "Coverage and Field Estimation on Bounded Domains by Diffusive Swarms." In *IEEE Conference on Decision and Control (CDC)*, pp. 2867–2874, 2016.

[EB15] K. Elamvazhuthi and S. Berman. "Optimal control of stochastic coverage strategies for robotic swarms." In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1822–1829, 2015.

[EN99] Klaus-Jochen Engel and Rainer Nagel. *One-parameter semigroups for linear evolution equations*, volume 194. Springer Science & Business Media, 1999.

[Fat99] H. O. Fattorini. *Infinite Dimensional Optimization and Control Theory*, volume 54. Cambridge University Press, 1999.

[FS14] M. Fornasier and F. Solombrino. "Mean-field optimal control." *ESAIM: Control, Optimisation and Calculus of Variations*, **20**(4):1123–1152, 2014.

[GDG16] A. Griffiths, A. Dikarev, P. R. Green, B. Lennox, X. Poteau, and S. Watson. "AVEXIS – Aqua Vehicle Explorer for In-Situ Sensing." *IEEE Robotics and Automation Letters (RA-L)*, **1**(1):282–287, 2016.

[HBN13] Sabine Hauert, Spring Berman, Radhika Nagpal, and Sangeeta N Bhatia. "A computational framework for identifying design guidelines to increase the penetration of targeted nanoparticles into tumors." *Nano today*, **8**(6):566–576, 2013.

[HLM07] Seungpyo Hong, Pascale R Leroueil, István J Majoros, Bradford G Orr, James R Baker Jr, and Mark M Banaszak Holl. "The binding avidity of a nanoparticle-based multivalent targeted drug delivery platform." *Chemistry & biology*, **14**(1):107–115, 2007.

[HMG02] Adam T Hayes, Alcherio Martinoli, and Rodney M Goodman. "Distributed odor source localization." *IEEE Sensors Journal*, **2**(3):260–271, 2002.

[Hoe63] W. Hoeffding. "Probability inequalities for sums of bounded random variables." *Journal of the American Statistical Association*, **58**(301):13–30, 1963.

[HW08] H. Hamann and H. Wörn. "A framework of space–time continuous models for algorithm design in swarm robotics." *Swarm Intelligence*, **2**(2-4):209–239, 2008.

[KE11] Peter Kingston and Magnus Egerstedt. "Distributed-infrastructure multi-robot routing using a Helmholtz-Hodge decomposition." In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 5281–5286. IEEE, 2011.

[LFE17] Hanjun Li, Chunhan Feng, Henry Ehrhard, Yijun Shen, Bernardo Cobos, Fangbo Zhang, Karthik Elamvazhuthi, Spring Berman, Matt Haberland, and Andrea L Bertozzi. "Decentralized stochastic control of robotic swarm density: Theory, simulation, and experiment." In *Intelligent Robots and Systems (IROS)*, 2017.

[LL07] J.-M. Lasry and P.-L. Lions. "Mean field games." *Japanese Journal of Mathematics*, **2**(1):229–260, 2007.

[MB02] A. J. Majda and A. L. Bertozzi. *Vorticity and Incompressible Flow*, volume 27. Cambridge University Press, 2002.

[MCF13] K. Y. Ma, P. Chirarattananon, S. B. Fuller, and R. J. Wood. "Controlled flight of a biologically inspired, insect-scale robot." *Science*, **340**(6132):603–607, 2013.

[MH12] A. R. Mesquita and J. P. Hespanha. "Jump Control of Probability Densities With Applications to Autonomous Vehicle Motion." *IEEE Transactions on Automatic Control*, **57**(10):2588–2598, Oct. 2012.

[ML06] D. Milutinovic and P. Lima. "Modeling and optimal centralized control of a large-size robotic population." *IEEE Transactions on Robotics*, **22**(6):1280–1285, 2006.

[MWW07] Marcin Magdziarz, Aleksander Weron, and Karina Weron. "Fractional Fokker-Planck dynamics: Stochastic representation and computer simulation." *Physical Review E*, **75**(1):016708, 2007.

[OL13] A. Okubo and S. A. Levin. *Diffusion and Ecological Problems: Modern Perspectives*, volume 14. Springer Science & Business Media, 2013.

[Pav14] G. A. Pavliotis. *Stochastic Processes and Applications*, volume 60 of *Texts in Applied Mathematics*. Springer-Verlag New York, 2014.

[PCM11] A. Prorok, N. Correll, and A. Martinoli. "Multi-level Spatial Modeling for Stochastic Distributed Robotic Systems." *International Journal of Robotics Research*, **30**(5):574–589, 2011.

[PFB11] Marco Pavone, Emilio Frazzoli, and Francesco Bullo. "Adaptive and distributed algorithms for vehicle routing in a stochastic and dynamic environment." *IEEE Transactions on Automatic Control*, **56**(6):1259–1274, 2011.

[PRT15] B. Piccoli, F. Rossi, and E. Trélat. "Control to Flocking of the Kinetic Cucker–Smale Model." *SIAM Journal on Mathematical Analysis*, **47**(6):4685–4719, 2015.

[REB18] Ragesh K Ramachandran, Karthik Elamvazhuthi, and Spring Berman. "An optimal control approach to mapping GPS-denied environments using a stochastic robotic swarm." In *Robotics Research*, pp. 477–493. Springer, 2018.

[RF06] Raul Rojas and Alexander Gloye Förster. "Holonomic control of a robot with an omnidirectional drive." *KI-Künstliche Intelligenz*, **20**(2):12–17, 2006.

[RL15] C. Robin and S. Lacroix. "Multi-robot target detection and tracking: taxonomy and survey." *Autonomous Robots*, pp. 1–32, 2015.

[Roa] "A Roadmap for U.S. Robotics: From Internet to Robotics, 2013 Edition." `http://robotics-vo.us/sites/default/files/2013$%$20Robotics$%$20Roadmap-rs.pdf`. Robotics Virtual Organization.

[SRS09] Mac Schwager, Daniela Rus, and Jean-Jacques Slotine. "Decentralized, adaptive coverage control for networked robots." *The International Journal of Robotics Research*, **28**(3):357–375, 2009.

[TFL14] S. Tong, E. J. Fine, Y. Lin, T. J. Cradick, and G. Bao. "Nanomedicine: tiny particles and machines give huge gains." *Annals of Biomedical Engineering*, **42**(2):243–259, 2014.

[Tro10] F. Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, volume 112 of *Graduate Studies in Mathematics*. American Mathematical Society, 2010.

[TW08] Greg M Thurber and K Dane Wittrup. "Quantitative spatiotemporal analysis of antibody fragment diffusion and endocytic consumption in tumor spheroids." *Cancer research*, **68**(9):3334–3341, 2008.

[ZBE18] Fangbo Zhang, Andrea L Bertozzi, Karthik Elamvazhuthi, and Spring Berman. "Performance bounds on spatial coverage tasks by stochastic robotic swarms." *IEEE Transactions on Automatic Control*, **63**(6):1473–1488, 2018.