Chapter 1

Probabilistic Modeling of Swarming Systems: From Non-Spatial to Spatial Dynamics

Nikolaus Correll and Heiko Hamann

1.1 Introduction

The swarming behavior of ants, wasps and bees demonstrates the emergence of stupendously complex spatio-temporal patterns ranging from a swarm finding shortest paths to the assembly of three-dimensional structures with intrinsicate architecture and well-regulated thermodynamics [1, 2]. In the bigger scheme of things, these systems represent just the tip of the iceberg; their behavior is considerably less complex than that of the brain, cities or galaxies, all of which are essentially swarming systems (and all of which can be reduced to first principles and interactions on atomic scale). Yet, social insects make the world of self-organization accessible to us as they are comparably easy to observe. Studying these systems is interesting from an engineering perspective as they demonstrate how collectives can transcend the abilities of the individual member and let the organism as a whole exhibit cognitive behavior.

"Cognition" is derived from the Latin word *cognescere* and means "to know", "to recognize" and also to "conceptualize". In the human brain, cognition emerges — to the best of our knowledge — from the complex interactions of highly connected, large-scale distributed neural activity. We argue that "cognition" can manifest itself at multiple different levels of complexity ranging from conceptualizing collective decisions such as assuming a certain shape or deciding between different abstract choices in social insects to reasoning on complex problems and expressing emotions in humans, the combination of the latter two often framed as the "Turing test" in Artificial Intelligence. This chapter aims at developing formal models to capture the characteristic properties of the most simple cognitive primitives in swarming systems. In particular, we wish to understand the relationship between the activities of the individual member of the swarm and the dynamics that arise at collective level. The resulting models can be matched to data recorded from physical systems, be used to predict the outcome of a robot's individual behavior on a larger swarm, and used in an optimization framework to determine the best parameters that help improving a certain metric [4].

This chapter reviews probabilistic models of three swarming primitives that are examples of conceptualizations that are exclusively represented at the collective level: collaboration, collective decision making, and collective optimization. Guided by examples from social insects, we present models that generalize to arbitrary agent systems and can serve as building blocks for more complex systems. The probabilistic component of the models arises from (1) the agent's motion which often has a random component, (2) explicit random decisions made by individual agents, and (3) random encounters between agents. Randomness in an agent's motion can be introduced, for example, by physical properties such as slip, by deficits in robot hardware, or by explicitly explorative behavior, e.g., based on random turns. It is therefore reasonable to model at least the single-agent behavior with probabilistic methods. Yet, it is possible to model the expected swarm-level behavior using deterministic models. In such a swarm-level model the underlying stochastic motion of agents is summarized in macroscopic properties, which are averages such as the expected swarm fraction in a certain state or at a certain position [11, 26, 24, 20].

Such probabilistic models are in contrast with deterministic models of swarming systems, which explicitly model the positions of individual robots. Representative examples include controllers for flocking [17], consensus [25], and optimal sensor distribution for sampling a given probability density function [5]. While the robots' spatial distribution is explicitly modeled, those models have difficulties dealing with randomness or robot populations in which robots can be in different states at the same time.

After providing a brief background on phenomenological probabilistic models based on the master equation, this chapter will first review population dynamic models that ignore the spatial distribution of the individual robots and the swarm and then present models that explicitly model the spatial distribution of the robot swarm using time-dependent, spatial probability density functions.

1.2 Master equation

Let a robot be in a discrete set of states with probability $p_i \in \vec{P}$, with $\vec{P}$ a vector maintaining the probabilities of all possible states and $\sum \vec{P} = 1$. These states model internal states of the robot, determined by its program, or external states, determined by the state of the robot within its environment. Actions of the robot and environmental effects will change these probabilities. This is captured by a phenomenological set of first-order differential equations, also known as the master equation [30],

$$\frac{d\vec{P}}{dt} = \mathbf{A}(t)\vec{P} \tag{1.1}$$

where $\mathbf{A}(t)$ is the transition matrix consisting of entries $p_{ij}(t)$ that correspond to the probability of a transition from state i to state j at time t. Multiplying both sides with the total number of robots N_0, allows us to calculate the average number of robots in each state. For brevity, we write $N_i(t) = N_0 p_i(t)$. Similarly, when expanding the master equation for a continuous space variable, one finds the Fokker-Planck equation, also known as Kolmogorov Forward equation or the Smoluchowski equation [9, 29].

1.3 Non-spatial Probabilistic Models

1.3.1 Collaboration

An important swarming primitive is *collaboration*, which requires a number of agents (n) to get together at a site. Collaboration is different from the more general *task allocation* problem, in which the number of agents is not explicitly specified. In swarm robotics, a "site" can have spatial meaning, but can also be understood in an abstract way as means to form teams. Although there are many different algorithms for team formation, we focus on a collaboration mechanisms that was introduced in the "Stick-Pulling Experiment" [16] and turned out to be a recurrent primitive in swarm robotic systems, e.g., in swarm robotic inspection, where robots can serve as temporary markers in the environment [3]. Here, collaboration happens when an inspecting robot encounters a marker, which informs it that this specific area has already been inspected. The collaboration model therefore finds application in studying trade-offs between serving as "memory" to the swarm and actively contributing to the swarming behavior's metric.

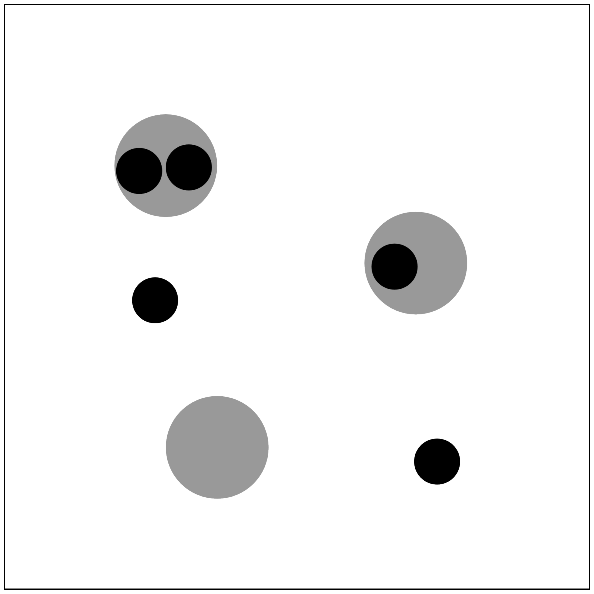


Figure 1.1: A collaboration example. $N_0 = 5$ robots (black) in a bounded environment with $M_0 = 3$ collaboration sites, each requiring $n = 2$ robots to be present simultaneously for collaboration to happen.

In the Stick-Pulling experiment N_0 robots are concerned with pulling M_0 sticks out of the ground

$n_s(t)$	Average number of searching agents
$n_w(t)$	Average number of waiting agents
p	Probab. to encounter/match a waiting agent
$c(t)$	Average rate of collaboration matches
N_0	Total number of agents
M_0	Total number of collaboration sites
T_w	Waiting time

Table 1.1: Notation used in the Collaboration model.

in a bounded environment. This task requires exactly $n = 2$ robots. Physically, this can be understood as a stick that is too long to be extracted from the ground by a single robot. Rather, every robot that grabs the stick can pull it out a little further and keeps it there until the next robot arrives. In this work, we abstract the classical stick-pulling experiment to a generic collaboration model in which robots are simply required to meet, see also Figure 1.1. Intuitively, the amount of time spent waiting for collaboration to happen is a trade-off between *(1)* waiting at a site to find a collaborator and *(2)* having the chance to find a collaborator oneself by actively browsing the environment. Finding the collaboration rate, and the individual parameters that lead to it, that is *optimal* for a given environment, i.e., the number of collaboration sites and the number of agents, illustrates how probabilistic models can be employed to design this process and finding optimal collaboration.

The following model is loosely based on the development in [23], which applies discrete time difference equations. For simplicity, we assume that collaboration happens instantaneously, and focus on a continuous-time representation and stochastic waiting times. Please refer to [23] for an extensive treatment of deterministic waiting times and [21] for an extension to $n > 2$ agents. Variables used in the equations that follow are summarized in Table 1.1.

Let $n_s(t)$ with $n_s(0) = N_0$ be the number of searching agents at time $t \in \mathbb{R}^+$ and N_0 the total number of agents. Let $n_w(t) = N_0 - n_s(t)$ be the number of waiting agents at time t. With p the probability to encounter or match a waiting agent and T_r the average time an agent will wait for collaboration, we can write

$$\begin{aligned} \dot{n}_s(t) &= -p(M_0 - n_w(t))n_s(t) \qquad (1.2) \\ &\quad +\frac{1}{T_r} n_w(t) \\ &\quad +pn_s(t)n_w(t) \qquad (1.3) \end{aligned}$$

Thus, $n_s(t)$ decreases by the rate at which searching agents encounter empty collaboration sites (of which exist $M_0 - n_w(t)$ at time t), and it increases by those agents that return either from unsuccessful (at rate $\frac{1}{T_r}$) or successful collaboration, i.e., find any of the $n_w(t)$ waiting agents.

In order to maximize the collaboration rate in the system we are interested in maximizing the rate at which robots return from successful collaboration, i.e., $c(t) = pn_s(t)n_w(t)$.

Solving for $\dot{n}_s(t) = 0$ and substituting $n_w(t) = N_0 - n_s(t)$ allows to calculate the number of robots at steady-state n_s^*:

$$n_s^* = \frac{(2N_0 - M_0)pT_r - 1 + \sqrt{8N_0pT_r(1 + (M_0 - 2N_0)pT_r)^2}}{4pT_r} \qquad (1.4)$$

As $n_w^* = N_0 - n_s^*$ by definition, we can write

$$c^* = p(n_s^* N_0 - {n_s^*}^2) \qquad (1.5)$$

The collaboration rate as a function of T_r and N_0 is shown in Figure 1.2. By solving $\frac{dc^*}{dn_s^*} = 0$, we can calculate $n_{s,opt}^* = \frac{1}{2}N_0$ that maximizes c^*. Substituting $n_{s,opt}^*$ into (1.4) and solving for T_r, we can calculate the optimal waiting time $T_{r,opt}$ as

$$T_{r,opt} = \frac{1}{(M_0 - N_0)p} \qquad (1.6)$$

As $T_{r,opt}$ cannot be negative, an optimal waiting time can only exist if $M_0 > N_0$. This intuitively makes sense, as if there are less agents than collaboration sites, waiting too long might consume all agents in waiting states. We can also see, that the more collaboration sites there are, the less an agent should wait. There are two interesting special cases: first, $N_0 = M_0$. In this case $T_{r,opt}$ is undefined. Considering that collaboration sites exceed agents by exactly one, $T_{r,opt}$ is fully defined by $1/p$. Thus, the higher the likelihood is that agents find a collaboration site, the lower the waiting time should be. In this case, it makes sense to release searching agents from wait states to find another agent to collaborate. If this likelihood is low, however, agents are better of waiting to serve as collaborators for few searching agents.

With $T_{r,opt}$ given by (1.6) we can derive the following guidelines for agent behaviors. First, an optimal wait time exists only if there are less agents than collaboration sites. Otherwise, longer waits improve the chance of collaboration. Second, if the number of agents, the number of collaboration sites and the likelihood to encounter a collaboration site are known to each agent at all times, e.g., due to global communication or shared memory, agents

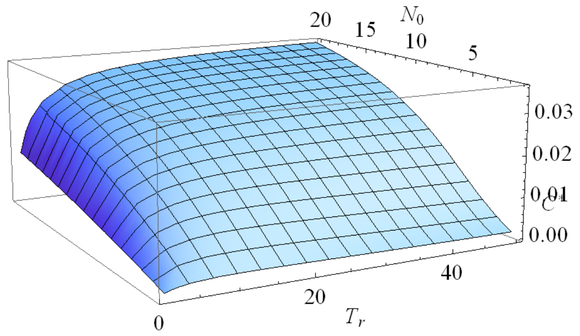


Figure 1.2: The collaboration rate as a function of T_r and N_0 for $M_0 = 10$ collaboration sites. There exists an optimal T_r for $N_0 < M_0$, whereas the collaboration rate increases steadily otherwise for increasing values of T_r.

could calculate $T_{r,opt}$ at all time. If these quantities are not known, however, agents could estimate these quantities based on their interactions in the environment, by observing the rates at which they encounter collaboration and empty sites. Individual agent learning algorithms that accomplish this goal are discussed in detail in [22].

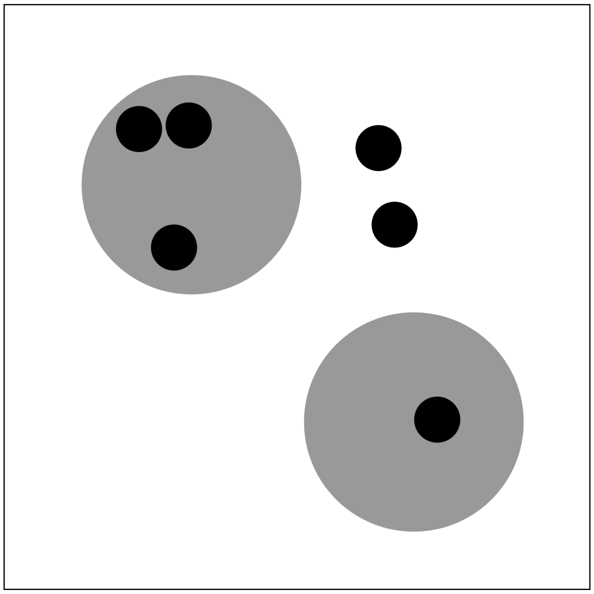


Figure 1.3: Collective decision example. $N_0 = 6$ robots decide between $M_0 = 2$ choices. Three plus one robot have already made decisions, two robots remain undecided.

$n_s(t)$	Average number of searching agents
$n_i(t)$	Average number of agents committed to choice i
p_i	Unbiased probability to select choice i
T_i	Unbiased time to stay with choice i
N_0	Total number of agents
M_0	Total number of choices
T_w	Waiting time

Table 1.2: Notation used in the collective decision model.

1.4 Collective Decisions

Another collective intelligent swarming primitive are *collective decisions*. These can be observed in path selection of ants [6], or shelter selection of cockroaches [10] or robots [8], but can also have non-spatial meaning, for example when a consensus on M_0 different discrete values is needed. An example of such a situation is shown in Figure 1.3. While the above references provide models that are specific to their application, this chapter provides a generalized model for collective decisions that rely on different ways of social amplification, i.e., a change of the behavior based on the activities of other swarm members, or the absence thereof.

Let $n_s(t)$ with $n_s(0) = N_0$ be the number of searching/undecided agents at time $t \in \mathbb{R}^{+}_{\not\perp}$ and N_0 the total number of agents. Let p_i, $0 < i \leq M_0$, be the unbiased probability for an agent to select value i from M_0 different values. This probability is unbiased as it does not depend on social amplification. We can then write the following differential equations for the number of agents $n_i(t)$ that have selected value i:

$$\dot{n}_i(t) = n_s(t)p_iR_i(t) - \frac{1}{T_i}n_i(t)Q_i(t), \quad n_i(0) = 0 \qquad (1.7)$$

$$n_s(t) = N_0 - \sum_{i=1}^{M_0} n_i(t) \qquad (1.8)$$

where T_i is the average time spent on solution i before resuming search, and $R_i(t), Q_i(t) : n_i(t), n_s(t) \rightarrow \mathbb{R}^+$ are functions that might or might not depend on the number of agents in other states, and therefore making the differential equation for $n_i(t)$ linear or non-linear, respectively. There are four interesting cases: both $R_i(t)$ and $Q_i(t)$ being constant, both being functions of one or more states of the system, e.g., $n_i(t)$ or $n_s(t)$, and combinations thereof.

In case both $R_i(t)$ and $Q_i(t)$ are constants, one can show that the number of agents selecting choice i at steady state n_i^* is given by

$$n_i^* = \frac{R_i}{Q_i} n_s^* \tag{1.9}$$

with n_s^* the number of agents that remain undecided at steady-state. (This results from agents discarding choices at rate $1/T_i$.) For example, for a two-choice system, using $n_s^* = N_0 - n_1^* - n_2^*$ yields the steady states

$$n_1^* = \frac{Q_2 R_1}{Q_1 Q_2 + Q_2 R_1 + Q_1 R_2} \tag{1.10}$$

$$n_2^* = \frac{Q_1 R_2}{Q_1 Q_2 + Q_2 R_1 + Q_1 R_2} \tag{1.11}$$

A solution for $R_1 = 0.01$ and $R_2 = 0.04$ and $Q_1 = Q_2 = 1/10$ is depicted in Figure 1.4 and leads to $\approx 7\%$ and $\approx 27\%$ of agents in states one and two respectively, while most agents remain undecided. In this system, the speed at which the steady-state is reached depends on the values of R_i, with higher values of R_i leading to faster decisions, whereas the steady-state of undecided agents is determined by Q_i, with lower values of Q_i corresponding to lower values of n_s^*. In particular, values for $Q_i = 1/100$ or $Q_i = 1/1000$ will drastically increase convergence, in this example to 67% and 78% for the majority choice, respectively.

In case $Q_i(t)$ is constant, but $R_i(t)$ is a non-linear function of the form $R_i(t) = f[n_i(t)]_i^\alpha$ with $\alpha_i > 1$ a constant, we observe $n_i(t)$ to grow faster due to *social amplification* of attraction; the larger $n_i(t)$ is, the larger is the positive influx into $\dot{n}_i(t)$. Systems with this property usually converge much faster than linear systems. For example, a system with

$$R_i(t) = \left(1 + \frac{n_i(t)}{N_0}\right)^{\alpha_i} \tag{1.12}$$

shows faster convergence than a linear system for $\alpha_i \geq 1$. Here, normalizing social attraction with N_0 provides independence of the dynamics of the number of agents. An example with $\alpha_i = 5$ is shown in Figure 1.4(b).

Similarly, in case $R_i(t)$ is constant, but $Q_i(t)$ is a non-linear function of the form $Q_i(t) = f[n_i(t)]^\beta$ with $\beta < 0$ a constant, we observe $n_i(t)$ to grow faster due to social amplification of rest; the larger $n_i(t)$, the smaller is the out-flux from $\dot{n}_i(t)$. For example, a system with

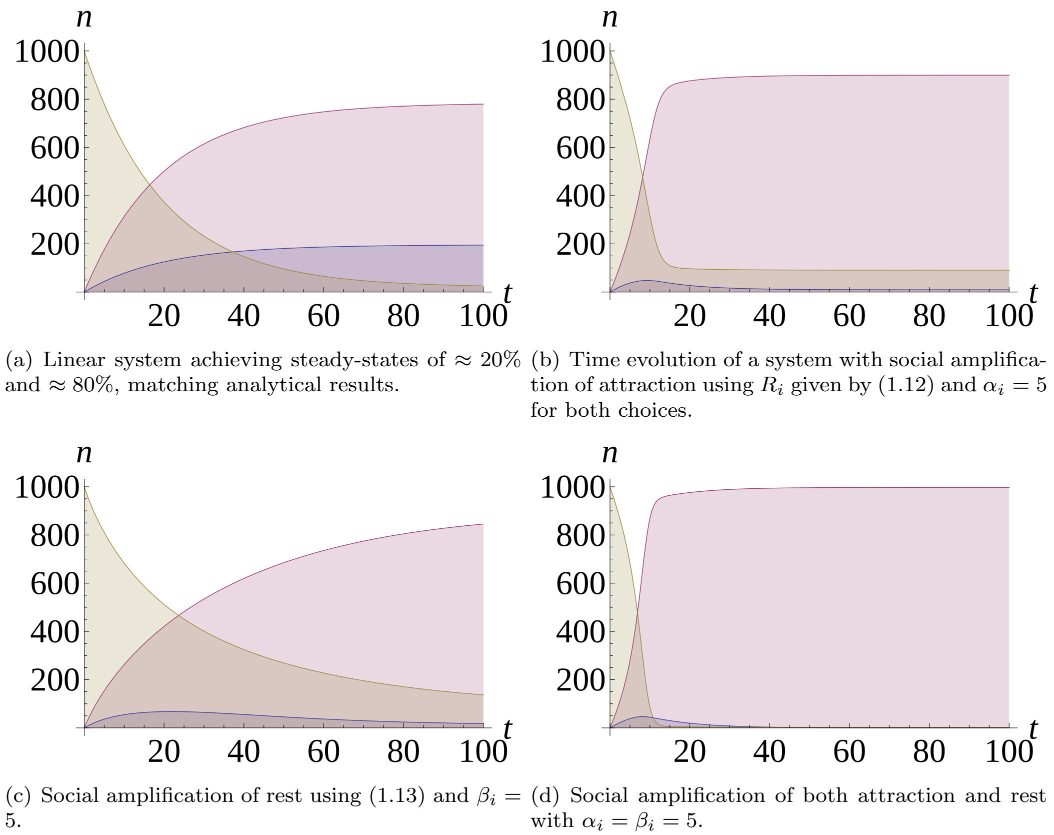
$$Q_i(t) = \left(1 + \frac{n_i(t)}{N_0}\right)^{\beta_i} \tag{1.13}$$

also shows faster convergence than a linear system. Notice that we do not consider positive exponents for β_i as this will drive agents away from decisions exponentially fast and will simply increase $n_s(t)$, i.e., the number of undecided agents. Results for a two-choice system with $\beta_i = 5$ is shown in Figure 1.4(c).

Finally, systems that rely both on social amplification of attraction *and* rest exhibit the best convergence, when compared with a purely linear system as well as systems that rely only on either social amplification mechanism. Results for a two-choice system with $\alpha_i = \beta_i = 5$ is shown in Figure 1.4(d).

Similar models, i.e., models that rely on non-linear amplification of either attraction, rest or both have been proposed for a series of social insect experiments. For example, in [6] an ant colony is presented with a binary choice to select the shortest of two branches of a bridge that connect their nest to a food source. Here, a model with social amplification of attraction — by means of an exponentially higher likelihood to chose a branch with higher pheromone concentration — is chosen and successfully models the dynamics observed experimentally. In [10] a model that uses social amplification of rest is chosen to model the behavior of a swarm of cockroaches deciding between two shelters of equal size but different brightness. The preference for cockroaches for darker shelters is expressed with a higher p_i for this shelter. Convergence to the dark shelter is then achieved by social amplification of rest, increasing the time cockroaches remain in a shelter exponentially with the number of individuals that are already in the shelter. Here, all cockroaches converge to a single shelter, even though the model proposed in [10] employs negative social amplification of attraction by introducing a notion of shelter capacity, which cancels the positive term in $\dot{n}_i(t)$ when the shelter reaches a constant carrying capacity. Finally, [18] presents a model for cockroach aggregation in which the likelihood to join an aggregate of cockroaches increases with the size of the aggregate, whereas the likelihood to leave a cluster exponentially decreases with its size.

The examples from the social insect domain are trade-offs between expressiveness of the model and its complexity. As the true parameter values of α_i and β_i are unknown, the same experimental data can be accurately matched by models with different dynamics. For example, social amplification of attraction as observed on larvae of German cockroaches in [18] was deemed to have negligible influence on mature American cockroaches selecting shelters with limiting carrying capacity in [10].



(a) Linear system achieving steady-states of ≈ 20% and ≈ 80%, matching analytical results.

(b) Time evolution of a system with social amplification of attraction using R_i given by (1.12) and $\alpha_i = 5$ for both choices.

(c) Social amplification of rest using (1.13) and $\beta_i = 5$.

(d) Social amplification of both attraction and rest with $\alpha_i = \beta_i = 5$.

Figure 1.4: Time evolution of a collective decision where solution two is picked four times as likely as solution one, and both solutions are re-evaluated after an average of 100s, for different non-linear dynamics. Graphs shows the fraction of agents picking solution one and two.

With respect to artificial agent and robotic systems, the presented models can instead provide design guidelines for achieving a desired convergence rate. At the same time, the models are able to support decisions on sensing and communication sub-systems that are required to implement one or the other social amplification mechanism.

1.5 Collective Optimization

The concept of optimization in collective systems is difficult to separate from the concept of collective decisions. Rather there seems to be a continuous transition. Collective decisions are made between several distinct alternatives implying a discrete world of options (e.g., left and right branch in path selection, two shelters etc.). Typically one refers to the term 'optimization' in collective systems in the case of tasks that allow for a vast (possibly even infinite) number of alternatives implying a continuous world of options.

For this optimization scenario we apply a probabilistic model that was reported before [11, 26, 14, 15, 27]. It is based on a stochastic differential equation (SDE, Langevin Equation) and a partial differential equation (PDE, Fokker–Planck Eq.) which can be derived from the former. While the Langevin Equation is a stochastic description of the trajectory in space over time of a single robot, the Fokker–Planck Equation describes the temporal evolution of the probability density in space for these trajectories. Hence, it can be interpreted as the average over many samples of robot trajectories (i.e., ensembles of trajectories). Even a second, more daring interpretation arises. We can interpret this probability density directly as a swarm density, that is, the expected fraction of the robot swarm for a given area and time. The deterministic PDE describes the mean swarm fraction in space and time. Interactions between robots can be modeled via dependence on the swarm density itself [11].

We introduce our formalism (see Table 1.3 for

$\mathbf{R}$	robot position
$\mathbf{A}$	direction & intensity of robots' directed motion
B	intensity of robots' random motion
$\mathbf{F}$	stochastic process (fluctuating directions)
$\mathbf{r}$	point in space
Q	theoretic term describing intensity of collisions
ρ_s	expected density of robots in state *stopped*
ρ_m	expected density of robots in state *moving*
w	waiting time
φ	rate of stopping robots

Table 1.3: Notation used in the Optimization model.

a summary of all used variables). The Langevin Equation which gives the position of a robot $\mathbf{R}$ at time t is

$$\dot{\mathbf{R}}(t) = -\mathbf{A}(\mathbf{R}(t), t) + B(\mathbf{R}(t), t)\mathbf{F}(t), \qquad (1.14)$$

whereas $\mathbf{A}$ defines directed motion via drift depending on the current position $\mathbf{R}$ and $B(\mathbf{R}(t), t)\mathbf{F}$ defines random motion based on $\mathbf{F}$ which is a stochastic process (e.g., white noise). Based on the Langevin Equation the Fokker-Planck Equation can be derived [30, 9, 7, 11]

$$\begin{aligned}\frac{\partial \rho(\mathbf{r}, t)}{\partial t} =& - \nabla(\mathbf{A}(\mathbf{r}, t)\rho(\mathbf{r}, t)) \\ &+ \frac{1}{2}Q\nabla^2(B^2(\mathbf{r}, t)\rho(\mathbf{r}, t)), \qquad (1.15)\end{aligned}$$

for a swarm density $\rho(\mathbf{r}, t)$ (according to the above interpretation) at position $\mathbf{r}$ and time t, a drift term $(-\nabla(\mathbf{A}(\mathbf{r}, t)\rho(\mathbf{r}, t)))$ due to directed motion and a diffusion term $(\frac{1}{2}Q\nabla^2(B^2(\mathbf{r}, t)\rho(\mathbf{r}, t)))$ due to random motion whereas typically we set $Q = 2$ for simplicity. According to our general approach [11] we introduce a Fokker–Planck Equation for each robot state and manage the transitions between states by rates similar to the rate equation approach of the above sections.

The optimization scenario considered here is inspired by the behavior of young honeybees. The algorithm, that defines the robots' behavior, is derived from a behavioral model of honeybees [28, 19]. Honeybees of an age of less than 24 hours stay in the hive, cannot yet fly, navigate towards spots of a preferred warmth of 36°C, and stay mostly inactive. An interesting example of swarm intelligent behavior is how they search and find the right temperature that their bodies need. It turns out that they do not seem to do a gradient ascent in the temperature field but rather a correlated random walk with inactive periods triggered by social interaction. Both the above mentioned behavioral model and the robot controller–called BEECLUST–are defined by the following.

```
1.) Each robot moves straight until it
    perceives an obstacle Ω within sensor
    range.
2.) If Ω is a wall the robot turns away and
    continues with step 1.
3.) If Ω is another robot, the robot
    measures the local temperature. The
    higher the temperature is the longer the
    robot stays stopped. When the waiting
    elapses, the robot turns away from
    the other robot and continues with
    step 1.
```

The temperature field, that we investigate in the scenario here, has one global optimum (36°C) at the right end of the arena and one local optimum (32°) at the left end of the arena. In analogy to the behavior observed in young honeybees the swarm is desired to aggregate fully at the global optimum but, at the same time, should also stay flexible within a possibly dynamic environment. The latter is implemented by robots (bees) that leave the cluster from time to time and explore the remaining arena. If a more preferable spot would emerge elsewhere they would start to aggregate there and the former cluster might shrink in size and finally vanish fully.

Now we apply the above modeling approach to this scenario. We have two states: *moving* and *stopped*. It turns out that in the *moving* state we do not have any directed motion, hence, we will turn the bias in the Langevin Eq. off (eq. 1.14, $\mathbf{A} = 0$). Without any directed motion in BEECLUST (no gradient ascent, actually movement fully independent from the temperature field) the Fokker–Planck Equation can be reduced to a mere diffusion equation in order to model the moving robots

$$\frac{\partial \rho(\mathbf{r}, t)}{\partial t} = \nabla^2(B^2(\mathbf{r}, t)\rho(\mathbf{r}, t)). \qquad (1.16)$$

This equation is our approach for state *moving* yet without addressing state transition rates.

The state *stopped* is even easier to model as it naturally lacks motion. That way it can be viewed as a reduction to a mere rate equation defined in each position $\mathbf{r}$. The state transition rates are defined by a stopping rate φ which can be determined, for example, empirically or by geometrical investigations (e.g., calculation of collision probabilities) [11]. For state *stopped* we obtain

$$\frac{\partial \rho_s(\mathbf{r}, t)}{\partial t} = \rho_m(\mathbf{r}, t)\varphi - \rho_m(\mathbf{r}, t - w(\mathbf{r}))\varphi, \quad (1.17)$$

for a stopping swarm fraction $\rho_s(\mathbf{r}, t)\varphi$ at spot $\mathbf{r}$ and time t and an awakening swarm fraction $\rho_s(\mathbf{r}, t - w(\mathbf{r}))\varphi$. The robots stop and wait for a time period $w(\mathbf{r})$ which depends on the temperature at spot $\mathbf{r}$.

Here, we choose to approximate the robots' correlated random walk as mere diffusion in a rough estimation. The function B in eq. 1.16 is reduced to a diffusion constant D. We add the rates of stopping/awakening and obtain the equation for state *moving*

$$\begin{aligned}\frac{\partial \rho_m(\mathbf{r}, t)}{\partial t} =& D\nabla^2 \rho_m(\mathbf{r}, t) - \rho_m(\mathbf{r}, t)\varphi \\ &+ \rho_m(\mathbf{r}, t - w(\mathbf{r}))\varphi.\end{aligned} \quad (1.18)$$

If we ignore diffusion and focus on one point in space we would have a mere rate equation similar to the above sections (except for the time-delay):

$$\dot{\rho}_m(t) = -\rho_m(t)\varphi + \rho_m(t - w(\mathbf{r}))\varphi. \quad (1.19)$$

Using eq. 1.18 (eq. 1.17 is mathematically not necessary) we can model the BEECLUST behavior. For a provided initial distribution of the robots we end up with an initial value problem for a PDE which we can solve numerically. The solution of this initial value problem is the temporal evolution of the swarm density. In Fig. 1.5 we compare the model to the results obtained by a simple agent-based simulation of BEECLUST. This comparison is meant to be qualitative only. The model catches most of the qualitative features that occur in simulation although we approximate the robots' motion in a rough estimation by diffusion.

Our approach shows how borders between the fields of engineering and biology vanish in swarm robotics. The BEECLUST algorithm is at the same time a controller for robots but also a behavioral model of an animal. The same Fokker–Planck model is used to model the macroscopic behavior of honeybees and robot swarms.

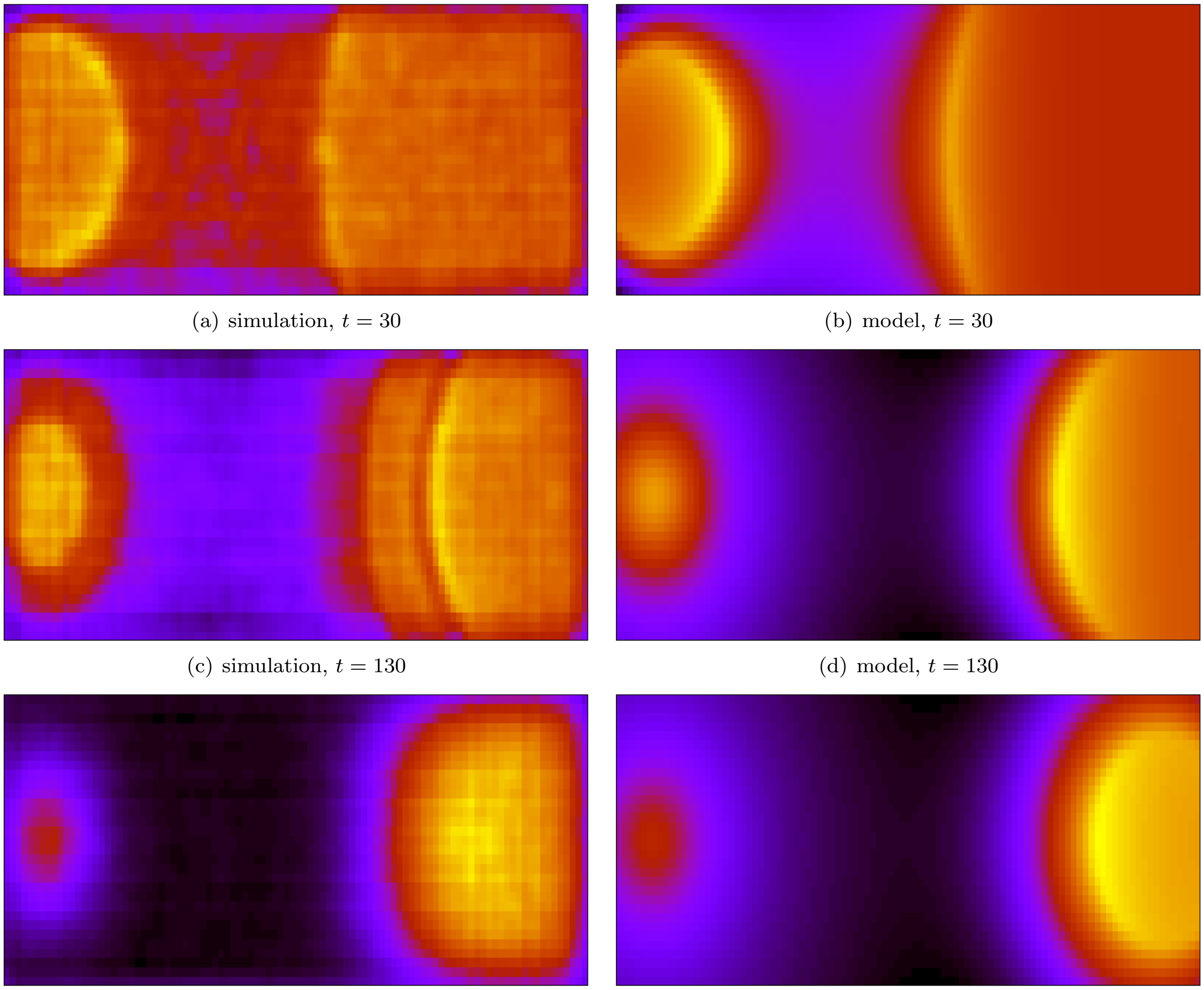
The Fokker–Planck model gives good estimates for expected swarm densities in space, the transient/asymptotic behavior of the swarm, and density flows. Modeling space explicitly allows for specific investigations such as objective areas and obstacles of certain shapes. Other case studies included an emergent taxis task which relies on one group of robots that is 'pushing' another group by collision avoidance [11], a collective perception task in which robots have to discriminate aggregation areas of different sizes [13], and a foraging task [12]. This model is mostly relevant to scenarios with spatially inhomogeneous swarm densities, that is, swarms forming particular spatial structures that cannot be averaged over several runs.

1.6 Conclusion

We presented mathematical models for three distributed swarming behaviors: collaboration, deciding between different choices, and optimization. Each of these processes are collective decisions of increasing complexity.

While the behaviors and trajectories of individual robots might be erratic and probabilistic, the average swarm behavior might be considered deterministic. This holds for both the models and the observed reality in robotic and biological experiments. An analogy is the distinction between the complex, microscopic dynamics of multi-particle systems and the much simpler properties of the corresponding ensembles of such systems in thermodynamics. This insight is important as it allows us to design the individual behavior so that the expected value of collective performance is maximized.

Although we presented models with increasing level of spatiality — from collaboration sites in the environment to modeling the distribution of robots over continuous space — modeling swarming systems with heterogeneous spatial and state distributions using closed form expressions is still a major challenge. Better understanding swarming systems with non-uniform spatial distributions will help us to better understand the impact of environmental patterns such as terrain, winds or current, thereby enabling swarm engineering for a series of real world applications that swarming systems have yet to tackle.



(a) simulation, $t = 30$

(b) model, $t = 30$

(c) simulation, $t = 130$

(d) model, $t = 130$

(e) simulation, $t = 200$

(f) model, $t = 200$

Figure 1.5: Comparison of histograms of swarm density obtained by an agent-based simulation and the corresponding model based on eq. 1.18 for different times and an initial state with equal distribution of robots. An optimal temperature peak of 36°C is at the right end of the arena, at the left end there is a suboptimal peak in temperature of 32°C, the middle part is cooler. We observe that in average at first clusters form at both ends of the arena but later those at the left vanish. Swarm size is $N = 25$. The histograms obtained by simulation are based on 10^6 samples.

Bibliography

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

[2] S. Camazine, J.-L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organization in Biological Systems.* Princeton Studies in Complexity. Princeton University Press, 2001.

[3] N. Correll and A. Martinoli. Modeling and analysis of beaconless and beacon-based policies for a swarm-intelligent inspection system. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2477–2482. IEEE, 2005.

[4] N. Correll and A. Martinoli. Towards optimal control of self-organized robotic inspection systems. In *8th Int. IFAC Symp. on Robot Control (SYROCO)*, Bologna, Italy, 2006.

[5] J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Automatic Control*, 20(2):243–255, 2004.

[6] J.-L. Deneubourg, S. Aron, S. Goss, and J. M. Pasteels. The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, 3:159–168, 1990.

[7] J. L. Doob. *Stochastic Processes.* Wiley, New York, 1953.

[8] S. Garnier, C. Jost, R. Jeanson, J. Gautrais, M. Asadpour, G. Caprari, J.-L. Deneubourg, and G. Theraulaz. Collective decision-making by a group of cockroach-like robots. In *2th IEEE Swarm Intelligence Symposium (SIS)*, 2005.

[9] H. Haken. *Synergetics - an introduction.* Springer-Verlag, Berlin, Germany, 1977.

[10] J. Halloy, J.-M. Amé, G. S. C. Detrain, G. Caprari, M. Asadpour, N. Correll, A. Martinoli, F. Mondada, R. Siegwart, and J.-L. Deneubourg. Social integration of robots in groups of cockroaches to control self-organized choice. *Science*, 318(5853):1155–1158, 2009.

[11] H. Hamann. *Space-Time Continuous Models of Swarm Robotics Systems: Supporting Global-to-Local Programming.* Springer-Verlag, 2010.

[12] H. Hamann and H. Wörn. An analytical and spatial model of foraging in a swarm of robots. In E. Şahin, W. Spears, and A. F. T. Winfield, editors, *Swarm Robotics - Second SAB 2006 International Workshop*, volume 4433 of *LNCS*, pages 43–55, Berlin, Heidelberg, Apr. 2007. Springer.

[13] H. Hamann and H. Wörn. A space- and time-continuous model of self-organizing robot swarms for design support. In *First IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO'07), Boston, USA, July 9-11*, pages 23–31, Los Alamitos, CA, July 2007. IEEE Press.

[14] 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, Oct. 2008.

[15] H. Hamann, H. Wörn, K. Crailsheim, and T. Schmickl. Spatial macroscopic models of a bio-inspired robotic swarm algorithm. In *IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems (IROS'08)*, pages 1415–1420, Los Alamitos, CA, 2008. IEEE Press.

[16] A. J. Ijspeert, A. Martinoli, A. Billard, and L. Gambardella. Collaboration through the exploitation of local interactions in autonomous collective robotics: the stick pulling experiment. *Autonomous Robots*, 11:149–171, 2001.

[17] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on automatic control*, 48(6):988–1001, 2003.

[18] R. Jeanson, C. Rivault, J.-L. Deneubourg, S. Blanco, R. Fournier, C. Jost, and G. Theraulaz. Self-organized aggregation in cockroaches. *Animal Behavior*, 69:169–180, 2005.

[19] S. Kernbach, R. Thenius, O. Kornienko, and T. Schmickl. Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic swarm. *Adaptive Behavior*, 17:237–259, 2009.

[20] A. Kettler and H. Wörn. A framework for Boltzmann-type models of robotic swarms. In *Proc. of IEEE Symposium on Swarm Intelligence (SIS'11)*, pages 131–138. IEEE, 2011.

[21] K. Lerman, A. Galstyan, A. Martinoli, and A.-J. Ijspeert. A macroscopic analytical model of collaboration in distributed robotic systems. *Artificial Life*, 7(4):375–393, 2001.

[22] L. Li, A. Martinoli, and Y. Abu-Mostafa. Learning and Measuring Specialization in Collaborative Swarm Systems. *Adaptive Behavior*, 12(3–4):199–212, 2004.

[23] A. Martinoli, K. Easton, and W. Agassounon. Modeling of swarm robotic systems: A case study in collaborative distributed manipulation. *International Journal of Robotics Research*, 23(4):415–436, 2004.

[24] D. Milutinovic and P. Lima. *Cells and Robots: Modeling and Control of Large-Size Agent Populations.* Springer-Verlag, Berlin, Germany, 2007.

[25] R. Olfati-Saber and R. Murray. Consensus problems for networks of dynamic agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49:1520–1533, 2004.

[26] A. Prorok, N. Correll, and A. Martinoli. Multi-level spatial modeling for stochastic distributed robotic systems. *The International Journal of Robotics Research*, 30(5):574–589, 2011.

[27] T. Schmickl, H. Hamann, H. Wörn, and K. Crailsheim. Two different approaches to a macroscopic model of a bio-inspired robotic swarm. *Robotics and Autonomous Systems*, 57(9):913–921, September 2009.

[28] T. Schmickl, R. Thenius, C. Möslinger, G. Radspieler, S. Kernbach, and K. Crailsheim. Get in touch: Cooperative decision making based on robot-to-robot collisions. *Autonomous Agents and Multi-Agent Systems*, 18(1):133–155, February 2008.

[29] F. Schweitzer. *Brownian Agents and Active Particles. On the Emergence of Complex Behavior in the Natural and Social Sciences.* Springer-Verlag, Berlin, Germany, 2003.

[30] N. G. van Kampen. *Stochastic processes in physics and chemistry.* North-Holland, Amsterdam, 1981.