A Robust Model For Predicting Collective Behavior in Large Robot Swarms

John Harwell[1] and Angel Sylvester[1] and Maria Gini[1*]

Abstract

We study the forward collective behavior problem: how to predict swarm behavior given a problem description and high level characteristics of the robot control algorithm. We present a differential equation model of swarm behavior which does not require post-hoc parameter tuning or knowledge of the nature of the problem the swarm is working on to produce accurate predictions. Instead, our model computes its internal parameters directly from the problem description and robot control algorithm characteristics. We present results showing that our model accurately predicts behavior and performance in swarms of up to 12,000 robots across a wide range of object gathering scenarios.

1. Introduction

Swarm Robotics (SR) systems consist of many homogeneous robots [1] that perform tasks in a decentralized manner. Originally inspired from natural systems [2], SR systems have important properties [3] such as scalability, emergent self-organization, flexibility, and robustness. Collectively, these properties are the main attractive aspects of SR systems, in comparison with other approaches. Previous work has developed a measurement methodology for the properties of emergent self-organization and scalability [4, 3], and a powerful differential equation modeling paradigm for swarm collective behavior [5, 6, 7, 8].

Models in this paradigm operate on both the forward problem, i.e., predicting collective behavior from features of the control algorithm each robot runs [5], and the inverse problem, i.e., incorporating design constraints into algorithm design in order to produce a desired collective behavior [6, 9]. However, models are not very robust, and make simplifying assumptions such as (1) homogeneous agent distributions, (2) homogeneous environments (e.g., no obstacles and/or a completely visible arena), and (3) Markov/semi-Markov scenarios [6]. Moreover, they require post-hoc model fitting and/or experimentally defined parameters which are specific to the implementation of the algorithm under study, and contain "hard-coded" assumptions about the nature of the scenario they are applied to (such as arena geometry), limiting reuse. Nevertheless, many notable applications of various flavors of this methodology have appeared in the literature, demonstrating its practical utility: the stick pulling experiment [10], the "house hunting model" [9, 6], and ant-inspired models that collaborate with or without communication [11, 8].

We build on these works and develop a more robust model to solve the forward collective behavior problem. We contribute to the SR literature as follows. First, we eliminate 4/5 of the free parameters from previous work [5, 12, 13], greatly reducing the need for post-hoc model fitting, in the context of an object gathering task. Second, we consider scenario characteristics when deriving analytical expressions for the free parameters, encoding their effect within our derived expressions rather than our model, improving reusability. Third, we provide results demonstrating our model's robustness, its accurate predictions of swarm behavior and performance across a wide range of scenarios in swarms of over 12,000 robots; previous work only demonstrated accuracy in a single scenario with ≤ 50 robots.

2. Related Work

Development of effective strategies for a SR system can be accelerated by drawing inspiration from natural systems, such as ants and bees, due to their shared properties [1, 14]. SR systems are well suited when robustness and flexibility are key to success, such as space. tracking lake health, clearing space in mining, hazardous material cleanup, search & rescue [15, 1, 14, 2].

Many SR systems use heuristic decision making, rather than combining nature inspiration with mathematical grounding [16]. Nevertheless, heuristic approaches have been effective for robots to operate with

[*1] Department of Computer Science & Engineering, University of Minnesota, Minneapolis, MN 55455 {harwe006, sylve057, gini}@umn.edu

incomplete information and limited computing power.

To develop models of swarm behavior, SR researchers have used averaging of large numbers of simulation runs to obtain empirically bounded insights into real-world problems [4]. However, the emphasis on empirical rather than rigorous mathematical models, has been the chief impediment to a wider use of SR systems; systematically varying individual agent parameters to study their effect on collective swarm behavior is impractical, even in simulation. Mathematical characterization of collective swarm behavior is much more difficult, but provides the means to precisely predict *a priori—without* the need of repetitive experiments.

In a swarm S of N robots each running a control algorithm κ, robots might need to respond to environmental signals that arrive at unpredictable times; such systems are well-modeled as asynchronous, and a macroscopic-continuous differential equation modeling approach for the *average* behavior of S in the steady state is appropriate [6]. By representing S as a differentiable, continuous quantity, its dynamics can be modeled with differential equations whose variables are the population counts associated with different roles. We emphasize that when using macroscopic differential equation models to determine behavior of S, it is possible that actual system behaviors are far from the average [17]. Usually, the larger the system, the smaller the fluctuations; the master equation [18] can be used to calculate the deviation from the average, but such calculations are often intractable, or algebraically difficult.

A promising mathematically rigorous methodology utilizing macro- and microscopic models for group dynamics and individual behavior over time has been developed [5, 6, 7, 8]. It uses (1) differential equations to model the behavior of the *average* number of robots in S in a given state, (2) discrete difference equations to model the stochastic transitions between robots states, and (3) stochastic simulation of discrete difference equations to compute state transition rates for all the robots. It draws on implementations of the stochastic master equation in chemistry and statistical physics [18], which is typically used to model expected average behavior of systems. Through the usage of rate constants and population fractions in each state, it is possible to mathematically assess the general behavior of a system under a variety of stochastic circumstances. Most importantly, this approach has predictive control and performance guarantees—crucial components to translating laboratory models into viable real-world solutions without needing simulation experiments.

We study this modeling paradigm in a *foraging task* where robots gather blocks across a finite operating arena and bring them to a central location (nest) under various conditions and constraints. Foraging is one of the most studied applications of SR, due to its straightforward mapping to real-world applications [14].

3. Background: Previous Modeling Work

Previous work using Ordinary Differential Equations (ODEs) from [5, 12] defines equations for $\frac{dN_s(t)}{dt}$, $\frac{dN_h(t)}{dt}$, $\frac{dN^s_{av}(t)}{dt}$, and $\frac{dB(t)}{dt}$. In our model, shown later in Section 5.1, we simplify the original equations and replace some parameters with mathematical derivations. The quantities modeled are listed in Table 1.

Quantity	Description
α_r, α_b	Robot encounter rate of blocks or robots
τ_h, τ_{av}	Mean robot homing time, and mean time spent avoiding collision
$N_h(t)$	Mean robots returning to nest with blocks
$N_s(t)$	Mean number of robots searching for blocks
$N^h_{av}(t)$	Mean number of robots avoiding collision while homing
$N^s_{av}(t)$	Mean number of robots avoiding collision while searching
$B_j(t)$*	Mean number of blocks in area j of arena
$\alpha_{r'}$^	Robot encounter rate near the nest
$B(t)$^	Mean number of blocks in the arena

Table 1: Summary of ODE model components. Component with a * is only in our model, components with a ^ are only in previous work.

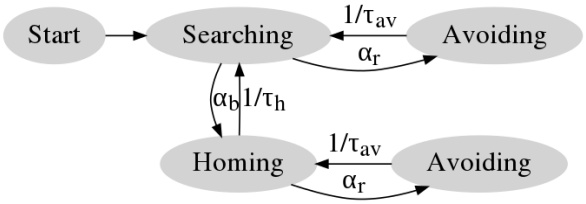


Figure 1: State diagram for a single robot. The *Avoiding* state is duplicated to uniquely identify the collision avoidance context: *Homing* or *Searching*. Transition rates are described in Table 1.

Each of the high level states in Fig. 1 maps directly to a single or a set/sequence of robot behaviors. We use a coarse-grained approach, starting from the *minimum* number of states we believe are sufficient to describe the system dynamics. Such an approach is generally more mathematically tractable, and if the results of the analysis do not sufficiently agree with the observed collective

behavior, then additional states can be added (see [5] for examples of model refinement by adding states).

4. Contributions

Our improved version of the model described in Section 3 has these differentiating characteristics:

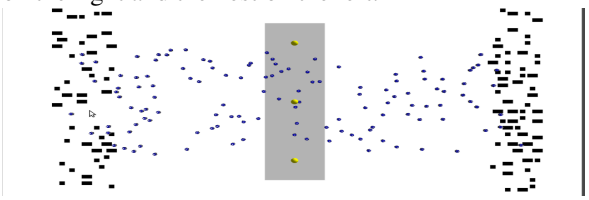
1. Structurally unbiased. We encode behavioral dynamics into our derivation of α_r, α_b, rather than encoding them in the model. This removes implicit knowledge of scenario characteristics and increases reusability of the model. For instance, if the nest is in a corner of the arena, the behavior of S near the nest is substantially different from the behavior further away; if we use this knowledge in the model, then it will have limited utility on dissimilar scenarios.

2. Fewer free parameters. Existing models have several free parameters, $\tau_h, \tau_{av}, \alpha_r, \alpha_r', \alpha_b$, computed via post-hoc model fitting. While they do not invalidate the theoretical basis of the models, they limit their reuse by requiring iterative parameter refinement. Instead we derive analytical expressions for all parameters except τ_{av} from scenario characteristics, greatly reducing the need for post-hoc model fitting; τ_{av} is directly computable from robot controller characteristics, or otherwise obtained via experiments with a single robot. We reduce the number of free parameters from five to one, and introduce the scenario collision avoidance characterization $L_{ca}()$, which allows us to extrapolate from the "base case" (RN), where S and objects are uniformly distributed, to other scenarios; it can be empirically determined without model fitting.

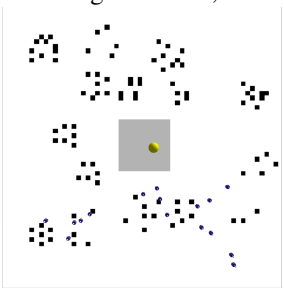
3. The spatial distribution of S and blocks is included in the model. Previous work assumes that S is uniformly distributed in 2D space, and that blocks are scattered randomly, hence the models are only accurate on *random* distributions (Fig. 2d). Random block distributions [8, 14]are appropriate in scenarios such as order fulfillment in a warehouse, but many scenarios cannot be modeled by such distributions. For example, transferring material from one side of a building to another location requires a *single source* (Fig. 2a) or *dual source* (Fig. 2b) block distribution model [19, 3]. For evacuation of civilians from a disaster zone, the block distribution cannot be inferred *a priori*, and a *power law* distribution in which blocks are clustered in groups of various sizes (Fig. 2c), is appropriate [14, 3]. Finally, we assume the number of blocks to be collected as infinite, which maps more naturally to a steady-state behavior.

By removing most free parameters and including the spatial distribution of S and blocks in our model, we guarantee that the model itself describes underlying characteristics of collective behavior, and is robust enough to be applied to a wide range of foraging scenarios. Our experimental results show that our model can accurately predict swarm behavior and performance in scenarios with the block distributions shown in Fig. 2 for swarms of up to 12,000 robots.

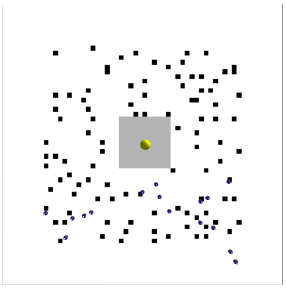
(a) *Single source* (SS) object distribution, with objects on the right and the nest on the left.



(b) *Dual source* (DS) object distribution, with objects on the right and left, and the nest in the center.



(c) *Power law* (PL) object distribution, with the nest in the center.



(d) *Random* (RN) object distribution, with the nest in the center.

Figure 2: Example foraging distributions.

5. Generalized Foraging Model

Our foraging model makes the following assumptions that enable it to make accurate predictions of collective swarm behavior in the steady state for each of the block distributions in Fig. 2:

1. S is homogeneous, and the control algorithm used by the robots is a Correlated Random Walk (CRW) [20, 3], which is a random walk where the direction of the next step is biased based on the direction of the previous step. The bias angle θ is drawn from a probability distribution $f(\theta)$ [21]. Robots do a CRW until they acquire a block, which they then transport to the nest using phototaxis (i.e., motion in response to light) at a constant

speed $\bar{v_h}$. Robots are reactive and have no memory.

2. To reduce congestion in the nest, robots do not return to the nest center to drop their carried object. Instead, they choose a random nest interior point along their trajectory to treat as the center; this shortens the distance traveled when homing by an easily computable amount.
3. Once deposited in the nest, any collected block is redistributed immediately in the operating area, and the overall number of blocks remains the same over time (there might be fewer blocks available in the arena when some are carried by robots).
4. S has reached steady state at some $t \gg 0$.
5. The swarm density ρ_S is relatively low, so the behavior of N robots is a linear function of the behavior of the swarm with $N = 1$ robots. An important consequence of this is that we ignore the effect of a robot avoiding collision and encountering *another* robot during its avoidance manuevers.

5.1. ODE Model

Let $M = \{SS, DS, RN, PL\}$ be the set of scenarios based on the block distributions shown in Fig. 2. In each $m \in M$, let the area where blocks can be distributed be a subset of the overall operating area A of the arena. Let $j = 1, \ldots, J$ be the sub-areas within A in which blocks can be distributed, each described by a tuple $(A_j, \mathbf{c}_j, \mathbf{d}_j, \rho_j)$. A_j is the area occupied by the sub-area j, $\mathbf{c}_j$ is the center of the sub-area, $\mathbf{d}_j$ is the dimension, and ρ_j is the mean steady state block density within the sub-area. Then, the distributable area within A is the union of these disjoint subsets: $A_d = \cup A_j$.

Our improved ODE model is in Eqs. (1) to (4). We simplified the original equations from [5, 12], removed α_r', and replaced $\alpha_b, \alpha_r, \tau_h$ with mathematical derivations. Eq. (1) describes the change in the number of robots in the searching state. It decreases as robots pickup blocks, or encounter other robots and switch to the collision avoidance state. It increases as homing robots deposit blocks in the nest, or as searching robots exit the collision avoidance state. Eq. (2) describes the change in the number of robots in the homing state. It increases as robots pickup blocks, or leave the collision avoidance state. It decreases as robots enter the collision avoidance state, or deposit their block in the nest. Eq. (3) describes the change in the number of robots avoiding collision with other robots.

Eq. (4) uses the described block modeling method to capture the underlying behavioral dynamics of the swarm. To derive it, we made the following additional assumptions about block distribution. First, whenever a block is redistributed all j are selected with probability proportional to the fraction of distributable area they contain. Second, blocks are distributed uniformly within a given j. Third, all j can hold any number of blocks, allowing for two blocks to occupy the same location (i.e., stacking). We note that under our steady state assumption, Eq. (4) can be solved analytically given $B(0)$, and does not need to be solved numerically.

$$\frac{dN_s(t)}{dt} = -\alpha_b - \alpha_r + \frac{1}{\tau_h}N_h(t) + \frac{1}{\tau_{av}}N^s_{av}(t) \quad (1)$$

$$\frac{dN_h(t)}{dt} = \alpha_b - \alpha_r - \frac{1}{\tau_h}N_h(t) + \frac{1}{\tau_{av}}N^h_{av}(t) \quad (2)$$

$$\frac{dN^s_{av}(t)}{dt} = \alpha_r - \frac{1}{\tau_{av}}N^s_{av}(t) \quad (3)$$

$$\frac{dB_j(t)}{dt} = \frac{1}{\tau_h}N_h(t)\frac{A_j}{A_d} - \alpha_b\frac{A_j}{A_d} \quad j = 1, \ldots, J \quad (4)$$

Next we derive analytical models for $\tau_h, \alpha_b, \alpha_r$ from scenario parameters: arena geometry, number of blocks, block distribution, etc. We do not derive τ_{av}, because it depends intrinsically on the interference avoidance strategy employed by S and cannot be derived independently from κ without additional assumptions.

5.2. Derivation of Homing Time τ_h

Since searching begins from the nest, the density of $N_s(t)$ must be *greater* near the nest, because robots perform *biased* random walks. Consequently, we expect that the mean distance from the nest at which a searching robot encounters a block is *not* the same as the mean distance of a block from the nest. The acquisition density should fall off linearly, with the falloff moderated by the block density within a given j, $\rho_j = B_j(t)/A_j$. We expect a slower rate of decay of block acquisition distance as a function of distance from the nest within A_j for *lower* ρ_j than for higher. We also expect that ρ_j would play an exponentially moderating role only when $\|\mathbf{x} - \mathbf{x}_n\|$ is small, such as for RN or PL block distributions. For SS and DS distributions, where the mean distance from a block to the nest is large, the effect of ρ_j on block acquisition locations should be minimal.

We formulate our block acquisition probability density function for a robot at location $\mathbf{x}$ in the arena as follows, modeling the nest as a single point $\mathbf{x}_n$. Our function (Eq. (5)) is a close approximation of the occupancy distribution of a single random walker performing a biased random walk [21]. C is a normalization constant to ensure our density integrates to 1 over all j.

$$p_{acq_j}(\mathbf{x}) = \frac{C}{\left(\sqrt{\|\mathbf{x} - \mathbf{x}_n\|} - \frac{\ln(\rho_j)}{2\rho_j}\right)^2} \quad (5)$$

Having defined the probability density function, we now derive the expected acquisition location by finding

the expected values of the marginal density functions in x by integrating Eq. (5) over all j:

$$E[x_{acq}] = \sum_J \int_x \int_y p_{acj_j}(\mathbf{x}) x dx dy \tag{6}$$

and similarly for y. We now write an expression for τ_h^1:

$$\tau_h^1 = \frac{\left\| \mathbf{x}_{E_{acq}} - \mathbf{x}_n \right\|}{\bar{v_h}} \tag{7}$$

where $\bar{v_h}$ is the phototaxis velocity of the S, specified in the input configuration, and $\mathbf{x}_{E_{acq}}$ is the expected distance from the nest. $\bar{v_h}$ is *not* a free parameter, because it can be obtained from controller input configuration. To derive τ_h, we note that under our assumption of low to moderate ρ_S, the homing time increases linearly with N [5]. Specifically τ_h increases linearly with the expected value of time lost due to inter-robot interference, averaged across all robots:

$$\tau_h = \tau_h^1 \left[1 + \frac{\alpha_r \tau_{av}}{N} \right] \tag{8}$$

5.3. Derivation of Block Acquisition Rate α_b

For large N, S can be well-approximated as a fluid composed of robot particles, and obeys many of the same laws. Using this intuition, we can compute α_b by computing the mean time it takes a robot starting at the center of the nest to diffuse within A an RMS displacement of $\left\| \mathbf{x}_{E_{acq}} - \mathbf{x}_n \right\|$ to the expected acquisition location. Using the relationship between RMS displacement and diffusion, we have:

$$\frac{1}{\alpha_b} = \frac{\left\| \mathbf{x}_{E_{acq}} - \mathbf{x}_n \right\|^2}{2F(N)} \tag{9}$$

where $F(N)$ is the diffusion constant for a swarm of N CRW robots. An exact calculation of $F(N)$ is out of the scope of this paper, but we can approximate it as shown in Eq. (10) for uniform, symmetric scenarios (RN), and as $F(N)/\sqrt{2}$ otherwise; less interaction near the nest leads to slower diffusion. Using the results of the RMS drift for CRW [21], and our intuition that smaller θ will result in quickly swarm diffusion (more straight line motion), we have:

$$F(N) = N \frac{\sqrt{2} D_{xy}}{D_\theta} \tag{10}$$

The drift is characterized by $f(\theta)$ and the robot searching velocity v_s (a scenario parameter) as:

$$D_{x,y} = \frac{{v_s}^2}{4t} \underbrace{\int_{-\pi}^{\pi} (1 \pm \cos 2\theta) f(\theta) d\theta}_{D_\theta} \tag{11}$$

from which we can obtain D_{xy} as the D2 norm of Eq. (11).

5.4. Derivation of Robot Encounter Rate α_r

To derive α_r, we note that under our assumption of low to moderate ρ_S, the robot encounter rate will be a function of α_r^1 and $F(N)$. We view Fig. 1 as a queueing network, where robots are either performing collision avoidance maneuvers or not. The summed input/output transition rates for a state form the arrival and service rates for the collision avoidance queue Q_{ca} ($\lambda_{ca} = \alpha_r^1$ and $\mu_{ca} = 1/\tau_{av}$, respectively). Modeling Q_{ca} as a M/M/1 queue, (at most one robot exits collision avoidance per Δt, which is reasonable if Δt is small), we can write the following using Little's Law [22] as:

$$\alpha_r = \frac{N_{av}(t)}{\tau_{av}} - \alpha_r^1 N_{av}(t) \tag{12}$$

The second term in Eq. (12) is a corrective factor accounting for robots experiencing interference due to encountering arena walls, *not* other robots, which is simply the scaled rate at which a single robot experiences interference. We cannot use $N_{av}(t)$ directly, as α_r needs to be computed *a priori*, but we can estimate it as a function of α_r^1 and $N_{av}^1(t)$, using our intuition regarding swarm diffusion:

$$\hat{N_{av}}(t) = N_{av}^1(t) \frac{F(N)}{D_\theta} L_{ca}(m) \tag{13}$$

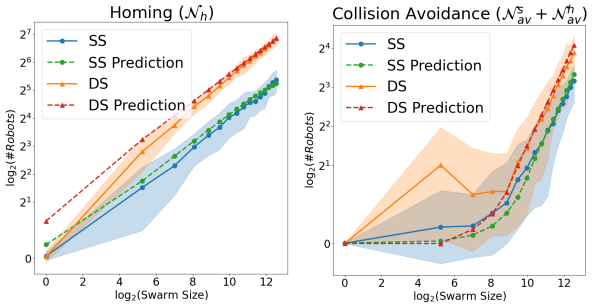
α_r^1 and $N_{av}^1(t)$ are calculable from κ (using the results of [21]), and are omitted here for brevity. We increase the influence of θ in Eq. (13) by introducing another divisive D_θ; smaller θ will result in more inter-robot interference due to straight line motion. $L_{ca}(m)$ characterizes the sub- or super-linearity of Eq. (13) for a scenario m in relation to the random (RN) scenario (see Table 2).

6. Experimental Framework

We employ a dynamical physics model of the marXbot robot in a 3D space for maximum fidelity (robots are still restricted to motion in the XY plane) using the ARGoS simulator [23] [1].

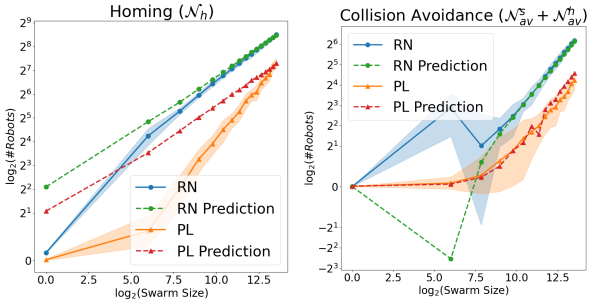
For all experiments we average the results of 32 runs of $T = 200{,}000$ seconds. We run our experiments with a constant $\rho_S = 0.01$=1 robot/$100m^2$, so that as we increase N, the size of the arena increases proportionally and the level of inter-robot interactions remains the same. We run our experiments in environments with single and dual source block distributions for $N = 1 \ldots 6{,}000$, and $N = 1 \ldots 12{,}000$ for random and power law block distributions. We use $\theta = \frac{\pi}{36}$

[1] Our code is at https://github.com/swarm-robotics/fordyca, https://github.com/swarm-robotics/sierra-plugin-fordyca



(a) SS and DS $N_h(t)$. (b) SS and DS $N_{av}(t)$.

Figure 3: Predictions of swarm behavior: $N_h(t)$ and $N_{av}(t)$ in single source (SS) and dual source (DS) scenarios across $N = 1 \ldots 6,000$.



(a) RN and PL $N_h(t)$. (a) RN and PL $N_{av}(t)$.

Figure 4: Predictions of swarm behavior: $N_h(t)$ and $N_{av}(t)$ in random (RN) and power law (PL) scenarios across $N = 1 \ldots 12,000$.

and the scenario collision avoidance characterizations shown in Table 2. For all scenarios but RN, we characterize them to follow our intuitions: DS scenarios exhibit comparatively superlinear interference, due to robots moving towards the central nest from opposing sides of the arena, contrary to SS scenarios, PL scenarios exhibit sublinear interference in comparison with RN scenarios due to the spacing of the clusters from the nest.

Scenario (m)	$L_{ca}(m)$	Scenario (m)	$L_{ca}(m)$
RN (random)	1	PL (power law)	$\frac{L_{ca}(RN)}{4\sqrt{2}}$
SS (single source)	$\frac{L_{ca}(RN)}{2\sqrt{2}}$	DS (dual source)	$2L_{ca}(RN)$

Table 2: Relative linearity factors used for the robot interference rate ($L_{ca}(m)$) of all scenarios in relation to RN, which is the "base case".

7. Results and Discussion

For all scenarios, we omit predictions for $N_s(t)$, since $N_s(t) = N - N_{av}(t) - N_h(t)$. We see strong agreement between our ODE model and experimental data for dual source (Fig. 3 red/orange) and single source scenarios (Fig. 3 blue/green) for $N_h(t)$, showing that it can cope with both symmetric (dual source) and asymmetric (single source) environments. Our model's prediction of behavior for $N = 1$ is off, but that it is close enough to capture the modeled collective dynamics accurately. RN scenarios are slightly less favorable than either SS or DS scenarios, because they do not have easily exploitable block clusters, though they are symmetric. Our model struggles to predict $N_h(t)$ with the same accuracy at smaller scales, though it converges to an accurate estimate at large N (Fig. 4(a) blue/green). This is likely due to the inaccuracy with $N = 1$ mentioned above, which becomes less consequential as interactive dynamics dominate at large N. Our model is likewise inaccurate at small N when predicting $N_{av}(t)$ for random distributions, but this is likely due to outliers in our simulation runs skewing the experimental data; we would expect a more gradual upward curve than an aberrant spike with $N \sim 60$.

PL scenarios are the least favorable of all foraging environments (asymmetric and without easily exploitable large block clusters); our model struggles to predict $N_h(t)$ (Fig. 4 red/orange) within the 95% confidence interval, and does not track the overall trend closely. However, it captures the dynamics of inter-robot interference, $N_{av}(t)$, accurately, showing that our underlying diffusion model and assumptions about linearity of α_r are generally accurate. This suggests that our heuristic block acquisition probability function produces inaccurate predictions of $N_h(t)$.

The accuracy of our model across scales and block distributions demonstrates that we are modeling the underlying behaviors in S. Given these results, accurate prediction of swarm performance scalability and emergent self-organization using [4] is possible.

8. Conclusions and Future Work

We have presented a robust ODE model for the forward collective behavior problem in foraging swarms, and shown it is accurate in a variety of scenarios in swarms of up to 12,000 robots, demonstrating its utility as an effective modeling tool which does not require extensive parameter tuning. Future work will derive the block acquisition density function using random walks theory and explore a solution to the inverse collective behavior problem: how to derive a swarm control algorithm to suit a desired collective behavior.

Acknowledgments: We gratefully acknowledge the MnDRIVE initiative, the Minnesota Robotics Institute, the UofM Informatics Institute, and the Minnesota Supercomputing Institute for their support of this research.

References

[1] E. Şahin, "Swarm robotics: From sources of inspiration to domains of application," in *Swarm Robotics*, ser. LNCS 3342. Springer, 2005, pp. 10–20.

[2] T. H. Labella and M. Dorigo, "Division of labor in a group of robots inspired by ants' foraging behavior," *ACM Trans. on Autonomous and Adaptive Systems (TAAS)*, vol. 1, no. 1, pp. 4–25, Sept. 2006.

[3] J. Harwell, L. Lowmanstone, and M. Gini, "Demystifying emergent intelligence and its effect on performance in large robot swarms," in *Proc. Autonomous Agents and Multi-agent Systems (AAMAS)*, May 2020, pp. 474–482.

[4] J. Harwell and M. Gini, "Swarm engineering through quantitative measurement of swarm robotic principles in a 10,000 robot swarm," in *Proc. Twenty-Eighth Int'l Joint Conference on Artificial Intelligence, IJCAI-19*, Aug. 2019, pp. 336–342.

[5] K. Lerman and A. Galstyan, "Mathematical model of foraging in a group of robots: Effect of interference," *Autonomous Robots*, vol. 13, no. 2, pp. 127–141, 2002.

[6] S. Berman, Á. Halász, V. Kumar, and S. Pratt, "Algorithms for the analysis and synthesis of a bio-inspired swarm robotic system," in *Swarm Robotics*. Springer-Verlag Berlin Heidelberg, 2007, vol. LNCS 4433, pp. 56–70.

[7] A. Galstyan, T. Hogg, and K. Lerman, "Modeling and mathematical analysis of swarms of microscopic robots," in *Proc. IEEE Swarm Intelligence Symposium, SIS 2005*, 2005, pp. 209–216.

[8] K. Sugawara and M. Sano, "Cooperative acceleration of task performance: Foraging behavior of interacting multi-robots system," *Physica D: Nonlinear Phenomena*, vol. 100, no. 3-4, pp. 343–354, 1997.

[9] A. Halasz, M. Hsieh, S. Berman, and V. Kumar, "Dynamic redistribution of a swarm of robots among multiple sites," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007, p. 2320–2325.

[10] A. 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*, vol. 11, no. 2, p. 149–171, 2001.

[11] K. Sugawara, I. Yoshihara, K. Abe, and M. Sano, "Cooperative behavior of interacting robots," *Artificial Life and Robotics*, vol. 2, no. 2, pp. 62–67, 1998.

[12] K. Lerman, A. Galstyan, A. Martinoli, and A. Ijspeert, "A macroscopic analytical model of collaboration in distributed robotic systems," *Artificial Life*, vol. 7, no. 4, pp. 375–393, 2001.

[13] K. Lerman and A. Galstyan, "Macroscopic Analysis of Adaptive Task Allocation in Robots," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 2, Oct. 2003, pp. 1951–1956.

[14] J. P. Hecker and M. E. Moses, "Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms," *Swarm Intelligence*, vol. 9, no. 1, pp. 43–70, 2015.

[15] C. Rouff *et al.*, "Properties of a formal method for prediction of emergent behaviors in swarm-based systems," in *Proc. 2nd Int'l Conference on Software Engineering and Formal Methods*, 2004.

[16] J. Abbott and A. P. Engelbrecht, "Nature-inspired swarm robotics algorithms for prioritized foraging," in *Swarm Intelligence. ANTS 2014*, D. M. et al., Ed. Springer, Cham, Sept. 2014, vol. Lecture Notes in Computer Science, vol 8667. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-09952-1_23

[17] K. Lerman, A. Galstyan, T. Hogg, and H.-P. Labs, "Mathematical analysis of multi-agent systems," arXiv:cs/0404002 [cs.RO], pp. 1–42, 2004.

[18] N. Kampen, *Stochastic Processes in Physics and Chemistry*, revised and enlarged ed. Elsevier Science, Amsterdam, 1992.

[19] G. Pini, A. Brutschy, M. Birattari, and M. Dorigo, "Task partitioning in swarms of robots: Reducing performance losses due to interference at shared resources," in *Informatics in Control Automation and Robotics*, ser. LNEE 85. Springer, 2011, pp. 217–228.

[20] E. Renshaw and R. Henderson, "The correlated random walk," *Journal of Applied Probability*, vol. 18, no. 2, pp. 403–414, 1981. [Online]. Available: http://www.jstor.org/stable/3213286

[21] E. A. Codling, R. N. Bearon, and G. J. Thorn, "Diffusion about the mean drift location in a biased random walk," *Ecology*, vol. 91, no. 10, pp. 3106–3113, 2010. [Online]. Available: http://www.jstor.org/stable/20788135

[22] M. Šeda, J. Šedová, and M. Horký, "Models and simulations of queueing systems," in *Recent Advances in Soft Computing. Proc. 22nd Int'l Conf. on Soft Computing (MENDEL 2016)*, R. Matoušek, Ed. Springer, 2017, vol. 576.

[23] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, M. Birattari, L. M. Gambardella, and M. Dorigo, "Argos: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm Intelligence*, vol. 6, pp. 271–295, 12 2012.