Optimization of Stochastic Strategies for Spatially Inhomogeneous Robot Swarms: A Case Study in Commercial Pollination

Optimization of Stochastic Strategies for Spatially Inhomogeneous Robot Swarms: A Case Study in Commercial Pollination

Spring Berman, Radhika Nagpal, and Ádám Halász

***Abstract*— We present a scalable approach to optimizing robot control policies for a target collective behavior in a spatially inhomogeneous robotic swarm. The approach can incorporate robot feedback to maintain system performance in an unknown environmental flow field. We consider systems in which the robots follow both deterministic and random motion and transition stochastically between tasks. Our methodology is based on an abstraction of the swarm to a macroscopic continuous model, whose dimensionality is independent of the population size, that describes the expected time evolution of swarm subpopulations over a discretization of the environment. We incorporate this model into a stochastic optimization method and map the optimized model parameters onto the robot motion and task transition control policies to achieve a desired global objective. We illustrate our methodology with a scenario in which the behaviors of a swarm of robotic bees are optimized for both uniform and nonuniform pollination of a blueberry field, including in the presence of an unknown wind.**

I. INTRODUCTION

A *robotic swarm* is a system that would consist of hundreds or thousands of autonomous, relatively expendable robots with limited sensing, communication, and computational capabilities. This kind of system has the potential to perform tasks with a high degree of parallelism, redundancy, flexibility, and adaptability to dynamic, possibly hazardous environments. A key challenge in robotic swarms is the development of approaches to design robot control policies that can provably produce a specific macroscopic outcome which is robust to disturbances in the system.

Fully centralized control strategies for a robot collective can provide globally optimal solutions but are computationally infeasible for such enormous populations. We use the paradigm of a broadcast architecture [18] to ensure scalability of the control approach with the swarm population size while enabling guarantees on system performance. A supervisory agent computes parameters that govern the robots' behaviors and transmits them to the swarm, without requiring information about the individual robot activities. Each robot in the entire swarm or a large subset is identical, unidentified, and follows the same set of decentralized algorithms, which rely only on local information from sensors and/or communication without knowledge of the global system state.

Our main contribution in this work is a top-down approach to synthesizing robot control policies that are *optimized* for a particular global objective to be accomplished by a *spatially inhomogeneous* swarm, whose members are arbitrarily distributed throughout their environment. We consider systems in which the robots transition stochastically between tasks at constant rates and follow a velocity field while executing random motion that can be modeled as Brownian motion. We describe optimization strategies that are *open-loop*, which compute the control policies independently of robot measurements, and *closed-loop*, which use this feedback to adapt the control policies to an unknown bulk motion of the medium, such as wind or water, in which the robots operate.

Our approach relies on the development of an abstraction of the physical system, which is facilitated by the stochasticity of the robot behaviors. We employ a modeling technique from the stochastic simulation of reaction-diffusion chemical systems [9]. From this model, we derive the *macroscopic continuous model*, a set of ordinary differential equations (ODE's) that govern the time evolution of average swarm subpopulations in each cell of a discretization of the environment. The dimensionality of this model is independent of the swarm population size; hence, for a fairly coarse discretization, it is much faster to numerically solve the ODE's than to simulate individual robots. This makes the model suitable for use in a stochastic optimization technique as a tool for quickly predicting system performance under a certain set of parameters. A supervisory agent can use such a technique to optimize the model parameters for a target objective in terms of swarm subpopulations. When these parameters are mapped to the robot motion controllers and stochastic policies for task transitions, the swarm produces the desired collective behavior.

In prior work on this topic [3], we described an advection-diffusion-reaction partial differential equation (PDE) model of a spatially inhomogeneous swarm and discussed the mapping between its parameters and the robot controllers. Similarly, [11], [22] develop PDE models of swarms based on the Fokker-Planck equation; they do not address the problem of controller optimization. Existing methods of optimizing control policies for swarms of robots whose behavioral rules are stochastic, or can be modeled as stochastic, apply to problems of task allocation [2], [5], [16], [20] and robotic assembly and self-assembly [7], [13], [17] that do not incorporate spatial descriptions of swarms. Optimization of a spatial model is considered in [19] for the specific purpose of directing a swarm to a desired location.

As in [3], we apply our methodology to a scenario of interest for the Robobees project [1], whose objective is to develop a colony of insect-inspired micro air vehicles [23]. We address the problem of designing control policies for a

S. Berman and R. Nagpal are with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 (e-mail: {sberman,rad}@eecs.harvard.edu)

Á. Halász is with the Department of Mathematics, West Virginia University, Morgantown, WV 26506 (e-mail: halasz@math.wvu.edu)

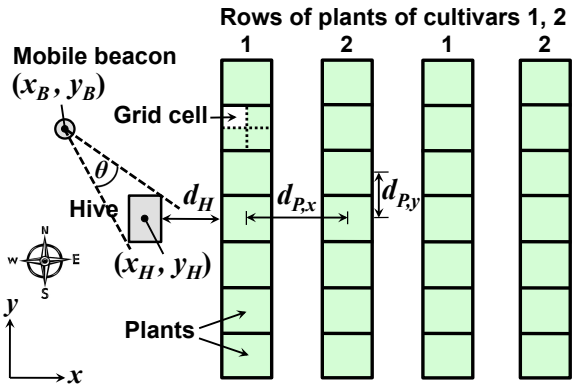

Fig. 1. A section of an orchard with a hive of robotic bees and a mobile beacon.

swarm of robotic bees that guarantee adequate pollination of an orchard of rabbiteye blueberries (*Vaccinium ashei*). We discuss the projected capabilities and constraints of the system in Section II and describe the models of the scenario in Section III. Section IV explains our optimization method, and Section V illustrates its use in producing specified distributions of pollination over an orchard, including in the presence of wind. More generally, this approach can be applied to any situation in which a swarm must achieve a desired number of repetitions of a task in different regions.

II. PROBLEM STATEMENT

A. Pollination objectives

We consider a scenario in which a swarm of robotic bees must pollinate a field with a given layout. The environment that we model is a section of a rabbiteye blueberry orchard, illustrated in Fig. 1, that consists of Q alternating rows of two cultivars. Insect pollination is needed for sufficient commercial yields of this crop, since almost all rabbiteye cultivars require cross-pollination with a compatible cultivar [6]. It is desirable to achieve uniform pollination of cultivated crops [8], but it can also be desirable to favor some cultivars or crop rows over others, depending on the crop timing.

To support this objective, we aim to design robot control policies that produce a uniform density of flower visits along rows of plants, and that can achieve any ratio between numbers of flower visits at plants of different cultivars. In addition, the pollination strategies should be able to incorporate feedback from the robots in order to fulfill the pollination task in the presence of unknown environmental disturbances, such as wind.

B. Robotic system components and capabilities

The components of our system are shown on the left side of Fig. 1. The robots would have sufficient power to undertake brief flights [12], or *sorties*, that originate from a location called the *hive*, and they would return to the hive to recharge. A computer at the hive can serve as the supervisory agent in our architecture. The computer calculates the parameters of the robot motion and task transitions for a specified pollination objective and transmits these parameters to the robots when they are docked at the hive for charging and uploading data. We assume that the robots can fly radially outward from a robotic *mobile beacon*, which initially spreads out the swarm from the hive to cover a broad area. To control the direction and angle θ of this swarm dispersal (see Fig. 1), the hive can command the location of the beacon before each sortie.

During a sortie, the robots are assumed to be capable of recognizing a flower that is very close by, potentially using an ultraviolet light sensor; flying to the flower; and hovering briefly while obtaining pollen from the flower using an appropriate appendage. In our scenario, a robot might vibrate a flower to release its pollen as is done by efficient blueberry pollinators [6]. Each robot is equipped with a compass and thus can fly with a constant heading. We assume that the robots can count the rows of plants that they fly over and switch between sets of parameters upon encountering a particular row.

In order to adapt the robot strategies to unforeseen disturbances, the hive must incorporate the spatial distribution of successful flower visits into its calculation of parameters for the next sortie. Hence, the robots must be able to record the locations of their flower visits and upload this information upon their return to the hive. A major constraint on our system is the lack of global positioning information, as it is infeasible to use GPS sensors on the tiny, highly power-constrained microrobotic platforms. We assume that the robots can perform discretized localization within a coarse grid over the field. A robot can identify the current grid cell it occupies using the angle of the beacon in a robot-fixed reference frame whose x-axis points east, as measured by the robot's compass. Alternatively, the robot may use dead reckoning combined with visual landmarks, such as the row edges. The robots record the number of flower visits they execute in each cell.

We note that any delay in swarm redeployment due to the data upload is not significant in our pollination application. For a realistic scenario with several thousand robots and a thousand cells, it is reasonable to estimate that the robots can simultaneously upload their visit counts to the hive over a wired medium in a few minutes; by comparison, rabbiteye blueberry flowers are open for 2-4 days [8].

C. Robot controller

We assume that the swarm performs a basic high-level behavior for pollination during a single sortie; the optimization method described in Section IV computes several of the parameters of this behavior over multiple sorties. Each member of a swarm of N robots performs the following actions during a sortie. Upon deploying from the hive, a robot flies radially away from the mobile beacon at a constant speed v, and once it encounters the edge of row $r_v \in \{1, ..., Q\}$, it begins to fly eastward at this speed. This switch in direction prevents the robots from flying beyond the domain to be pollinated and focuses the swarm on the plants. Concurrently with this deterministic motion, the robot

executes random movement that either arises from inherent noise due to sensor and actuator errors or is actively added by the robot's motion controllers, or both. This random motion provides a mechanism for the swarm to achieve thorough coverage of the field. We assume that the flowers are distributed densely enough such that a robot can always detect at least one flower in its sensing range when it flies over plants. While a robot is flying over row number j, it decides with probability per unit time $k_{hov,j}$ 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_{fly}, which determines the time taken to pollinate. A sortie lasts for a fixed time, t_s, at which the robot begins its return to the hive.

III. MODELS OF THE POLLINATION SCENARIO

A. Micro-Continuous Model

This model is used to simulate the robots' motion and probabilistic decisions to visit and leave flowers that are produced by the robot controller in Section II-C. A robot i has position $\mathbf{q}_i(t) \in \mathbb{R}^2$ at time t. While the robot is flying during a sortie, its deterministic motion is governed by the velocity field

$$\mathbf{v}(\mathbf{q}_i(t)) = \begin{bmatrix} v_x(\mathbf{q}_i(t)) \\ v_y(\mathbf{q}_i(t)) \end{bmatrix} = \begin{cases} v\frac{\mathbf{q}_i(t)-\mathbf{q}_B}{||\mathbf{q}_i(t)-\mathbf{q}_B||}, & x < x_{r_v}, \\ [v \;\; 0]^T, & x \geq x_{r_v}, \end{cases} \tag{1}$$

where $\mathbf{q}_B = [x_B \;\; y_B]^T$ are the mobile beacon coordinates and x_{r_v} is the x coordinate of the left boundary of row r_v. We model the random robot movement as a Brownian motion that drives diffusion, where D is the associated diffusion coefficient. D is the sum of D_{inh}, a constant determined by inherent noise, and D_{act}, a tunable parameter that produces actively added random motion. We assume that we can characterize D_{inh} and thus compute D. The displacement of robot i over each timestep Δt during flight is defined according to the Itō-Taylor integration scheme [9],

$$\mathbf{q}_i(t+\Delta t) - \mathbf{q}_i(t) = \mathbf{v}(\mathbf{q}_i(t))\Delta t + (2D\Delta t)^{1/2}\ \mathbf{Z}(t), \tag{2}$$

where $\mathbf{Z} \in \mathbb{R}^2$ is a vector of independent, normally distributed random variables with zero mean and unit variance. A robot is stationary while it is hovering.

The probabilistic decisions are modeled at each timestep Δt. When a robot is flying over row j, it decides to start hovering for pollination with probability $k_{hov,j}\Delta t$ at each timestep. A hovering robot resumes flying with probability $k_{fly}\Delta t$ per timestep.

B. Macro-Discrete Model

We develop an abstraction of the system in which the robots' motion and probabilistic decisions are modeled as chemical reactions, which form the basis for the model that we use in optimization. We discretize the environment into M square cells of side length l. If we use an *open-loop* optimization strategy, the minimum l is only constrained by the computational resources available at the hive to solve the corresponding macro-continuous model. If we use a *closed-loop* strategy, which requires data that is associated with each cell, then the minimum l is determined by the smallest region in which a robot can localize. The set $\mathcal{F}$ consists of the indices of the M_F cells that contain flowers. Sets $\mathcal{F}_{C1}$ and $\mathcal{F}_{C2}$ divide $\mathcal{F}$ according to the cultivar type of the flowers, and sets $\mathcal{F}_j$, $j \in \{1, ..., Q\}$, divide $\mathcal{F}$ according to the row number of the flowers. The set $\mathcal{H}$ contains the indices of the M_H cells that overlap with the hive.

We model the robots' motion through continuous space as a set of virtual reactions that represent transfers of robots between adjacent grid cells. This model relies on the approximation that the robots in each cell obey the well-mixed property, that is, they are uniformly randomly distributed throughout the cell. Borrowing from chemical reaction terminology, a *species* refers to the set of robots in a particular cell that are all performing a certain task. B^i_{fly} represents a member of the species that consists of robots that are flying in cell i. We define a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ for which the vertex set $\mathcal{V}$ contains the indices of the M cells and the edge set $\mathcal{E} = \{(i,j) \in \mathcal{V} \times \mathcal{V} \mid i \sim j\ \}$ contains the possible transitions of robots between pairs of neighboring cells, where $i \sim j$ denotes that a robot at cell i can move to cell j. Every edge (i,j) is associated with a rate constant k_{ij}, the probability per unit time for one robot in cell i to move to cell j. The set of reactions that describe the transfer of robots between cells is

$$B^i_{fly} \rightarrow^{k_{ij}} B^j_{fly}, \quad (i,j) \in \mathcal{E}. \tag{3}$$

We consider the environment to be unbounded. If i is a cell at the boundary of the domain under investigation, signified by $i \in \mathcal{B}_{in}$, and j is an adjacent cell that would exist on the other side of the boundary, signified by $j \in \mathcal{B}_{out}$, then we define a reaction to represent robots leaving the domain:

$$B^i_{fly} \rightarrow^{k_{ij}} \emptyset, \quad (i,j) \in \mathcal{E}_b, \tag{4}$$

where $\mathcal{E}_b = \{i \in \mathcal{B}_{in},\ j \in \mathcal{B}_{out} \mid i \sim j\ \}$.

Each rate constant is defined as the sum $k_{ij} = k^d_{ij} + k^v_{ij}$. The rate constant k^d_{ij} is the probability per unit time of robot transfer from cell i to j by diffusion and is defined as [9]

$$k^d_{ij} = D/l^2. \tag{5}$$

The rate constant k_{ij}^v converts deterministic motion governed by the velocity field (1) into a probability per unit time of transfer from cell i to j. Since we assume a uniform distribution of robots in each cell, this transfer probability in one timestep Δt, which we denote by $k_{ij}^{v1}\Delta t$, can be approximated as the area swept out in Δt by the cells' common edge moving with the velocity component in the direction from i to j, divided by the area of cell i, l^2. Dividing this expression by Δt yields

$$k_{ij}^{v1} = \frac{1}{l}\cdot\begin{cases}\max\left(v_x(x_i^c+\frac{l}{2},y_i^c),0\right), & \mathbf{q}_j^c=[x_i^c+l \;\; y_i^c]^T\\ \max\left(-v_x(x_i^c-\frac{l}{2},y_i^c),0\right), & \mathbf{q}_j^c=[x_i^c-l \;\; y_i^c]^T\\ \max\left(v_y(x_i^c,y_i^c+\frac{l}{2}),0\right), & \mathbf{q}_j^c=[x_i^c \;\; y_i^c+l]^T\\ \max\left(-v_y(x_i^c,y_i^c-\frac{l}{2}),0\right), & \mathbf{q}_j^c=[x_i^c \;\; y_i^c-l]^T,\end{cases} \tag{6}$$

where v_x, v_y are defined in equation (1) and $\mathbf{q}_i^c = [x_i^c \;\; y_i^c]^T$ is the center of cell i. For a swarm with no diffusive motion and no flower visits, the Euler method for integrating the corresponding macro-continuous model with $k_{ij}^v = k_{ij}^{v1}$ is equivalent to the donor-cell upwind finite volume method [15] for solving advection equations. To reduce the numerical diffusion that this method introduces, we observe that the upwind scheme can be written as the sum of central finite difference approximations to an advection term and a diffusion term with diffusion coefficients $k_{ij}^{v2}l^2$, where

$$k_{ij}^{v2} = \frac{1}{2l}\cdot\begin{cases}|v_x(x_i^c+\frac{l}{2},y_i^c)|, & \mathbf{q}_j^c=[x_i^c+l \;\; y_i^c]^T\\ |v_x(x_i^c-\frac{l}{2},y_i^c)|, & \mathbf{q}_j^c=[x_i^c-l \;\; y_i^c]^T\\ |v_y(x_i^c,y_i^c+\frac{l}{2})|, & \mathbf{q}_j^c=[x_i^c \;\; y_i^c+l]^T\\ |v_y(x_i^c,y_i^c-\frac{l}{2})|, & \mathbf{q}_j^c=[x_i^c \;\; y_i^c-l]^T.\end{cases} \tag{7}$$

We subtract out this diffusion by setting $k_{ij}^v = k_{ij}^{v1} - k_{ij}^{v2}$.

We also define reactions that model a robot's decisions to hover at a flower and to leave the flower after a pollination attempt. In these reactions, B_{hov}^i is a member of the species that consists of robots that are hovering at a flower in cell i, and V^i signals an instance of a flower visit in cell i:

$$B_{fly}^i \rightarrow^{k_{hov,j}} B_{hov}^i + V^i, \quad i\in\mathcal{F}_j,\; j\in\{1,...,Q\}, \tag{8}$$

$$B_{hov}^i \rightarrow^{k_{fly}} B_{fly}^i, \qquad i\in\mathcal{F}. \tag{9}$$

Let $N_s(t)$ be the integer population of species $s \in \{1,...,S\}$ in our system at time t. We define the vector of species populations,

$$\mathbf{N}(t) = [N_{B_{fly}^1}...N_{B_{fly}^M}|N_{B_{hov}^1}...N_{B_{hov}^M}|N_{V^1}...N_{V^M}]^T, \tag{10}$$

where each entry is evaluated at time t. The Chemical Master Equation [9] governs the probability of the system having any given set of species populations at each point in time. As we discuss next, we can use this equation to derive the *expected value* of $\mathbf{N}(t)$, which we will use as a prediction of the swarm behavior to evaluate the pollination metrics that serve as our optimization objective functions.

C. Macro-Continuous Model

This model consists of a set of ordinary differential equations that describe the time evolution of $\mu(t) \equiv \mathbb{E}(\mathbf{N}(t))$. We define the model using the quantities $\mathbf{D}_i \in \mathbb{R}^S$, the vector of species population changes that result from the execution of reaction $i \in \{1,...,R\}$, and $a_i(\mathbf{N})$, the *propensity* [10] associated with each reaction i. If the species population vector is $\mathbf{N}$ and reaction i occurs, this vector becomes $\mathbf{N}+\mathbf{D}_i$. The propensity $a_i(\mathbf{N})$ is defined such that $a_i(\mathbf{N})\Delta t$ is the probability that reaction i will occur in the next Δt, given a species population vector of $\mathbf{N}$. For a unimolecular reaction with reactant species j and rate constant k_i, $a_i(\mathbf{N}) = k_i N_j$. Since all of the reactions in our system are unimolecular, the dynamics of $\mu(t)$ can be derived as a closed set of ODE's from the Master Equation [10]:

$$\dot{\mu}(t) = \sum_{i=1}^{R}\mathbf{D}_i a_i(\mu(t)) \tag{11}$$

For the reaction network defined by (3), (4), (8), (9), equation (11) is

$$\begin{aligned}\dot{\mu}(t) &= \sum_{(i,j)\in\mathcal{E}}(\mathbf{u}_j-\mathbf{u}_i)\,k_{ij}\mu_i(t) - \sum_{(i,j)\in\mathcal{E}_b}\mathbf{u}_i k_{ij}\mu_i(t)\\ &+ \sum_{j=1}^{Q}\sum_{i\in\mathcal{F}_j}(\mathbf{u}_{i+M}+\mathbf{u}_{i+2M}-\mathbf{u}_i)\,k_{hov,j}\mu_i(t)\\ &+ \sum_{i\in\mathcal{F}}(\mathbf{u}_i-\mathbf{u}_{i+M})\,k_{fly}\mu_{i+M}(t).\end{aligned} \tag{12}$$

where $\mathbf{u}_i$ is the i^{th} Cartesian unit vector. This equation can be written in matrix form,

$$\dot{\mu}(t) = \mathbf{K}\mu(t), \tag{13}$$

where $\mathbf{K}\in\mathbb{R}^{3M\times 3M}$ contains the rate constants k_{ij}, $(i,j)\in\mathcal{E},\mathcal{E}_b$; $k_{hov,j}$, $j=1,...,Q$; and k_{fly}. Since equation (13) is a linear ODE model, its solution is

$$\mu(t) = e^{\mathbf{K}t}\mu(0). \tag{14}$$

IV. OPTIMIZATION

We use the macro-continuous model (13) to compute the tunable parameters of the system that maximize a metric of the degree of pollination over a specified number of sorties, β, each with duration t_s. We define a vector whose entries are the parameters to be optimized per sortie:

$$\mathbf{p} = [v \;\; D \;\; k_{hov,1} \;...\; k_{hov,Q} \;\; x_B \;\; y_B \;\; r_v]^T. \tag{15}$$

In general, v and D can be functions of row j in addition to the hovering probability rate, at the expense of increased computation time. Note that $\mathbf{K} = \mathbf{K}(\mathbf{p})$, and therefore $\mu(t) = \mu(\mathbf{p}, t)$ by equation (14). We denote the parameter vector for sortie j by $\mathbf{p}_j$ and define a matrix of parameters to be optimized for β sorties,

$$\mathbf{P} = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad ... \quad \mathbf{p}_\beta]. \tag{16}$$

The pollination metric is defined in terms of the *target number of flower visits* per cell $i \in \mathcal{F}$, $N^d_{V^i}$. One possible metric is the expected total fraction of pollination:

$$f_1(\mathbf{P}) = \frac{1}{M_F} \sum_{i \in \mathcal{F}} \min \left(\frac{\sum_{j=1}^{\beta} \mu_{V^i}(\mathbf{p}_j, t_s)}{N^d_{V^i}}, 1 \right). \tag{17}$$

Another metric, which we denote by $f_2(\mathbf{P})$, is the expected fraction of cells $i \in \mathcal{F}$ that receive at least $N^d_{V^i}$ visits. A third possible metric is a measure of distance from the target visit distribution:

$$f_3(\mathbf{P}) = -\sum_{i \in \mathcal{F}} \left| \sum_{j=1}^{\beta} \mu_{V^i}(\mathbf{p}_j, t_s) - N^d_{V^i} \right|. \tag{18}$$

The entries of the initial population vector $\mu(\mathbf{p}_j, 0)$ for each sortie j are defined as

$$\mu_{B^i_{fly}}(\mathbf{p}_j, 0) = \begin{cases} \lfloor N/M_H + 0.5 \rfloor, & i \in \mathcal{H},\ j = 1, ..., \beta, \\ 0, & i \notin \mathcal{H},\ j = 1, ..., \beta, \end{cases}$$

$$\mu_{B^i_{hov}}(\mathbf{p}_j, 0) = 0, \quad i = 1, ..., M, \quad j = 1, ..., \beta,$$

$$\mu_{V^i}(\mathbf{p}_j, 0) = \begin{cases} 0, & i = 1, ..., M,\ j = 1, \\ \nu_i(\mathbf{p}_{j-1}, t_s), & i = 1, ..., M,\ j = 2, ..., \beta. \end{cases}$$

In the *open-loop* strategy, $\nu_i(\mathbf{p}_{j-1}, t_s) = \mu_{V^i}(\mathbf{p}_{j-1}, t_s)$, *i.e.* the number of flower visits in a cell at the beginning of a sortie is set to the number at the end of the previous sortie as predicted by the macro-continuous model (13). In the *closed-loop* strategy, $\nu_i(\mathbf{p}_{j-1}, t_s)$ is the total number of visits to cell i in $j-1$ sorties that have been recorded by the robots and uploaded to the hive upon their return.

Defining the objective function as $f(\mathbf{P}) = f_1(\mathbf{P})$, $f_2(\mathbf{P})$, or $f_3(\mathbf{P})$, we can now formulate the optimization problem:

$$\begin{aligned} \text{maximize} \quad & f(\mathbf{P}) \\ \textit{subject to} \quad & \mu(\mathbf{p}_j, t_s) = e^{\mathbf{K}(\mathbf{p}_j) t_s} \mu(\mathbf{p}_j, 0), && (19) \\ & p_i^{min} \le \mathbf{P}_{i,j} \le p_i^{max},\ i = 1, ..., Q+4, && (20) \\ & \mathbf{P}_{(Q+5),j} \in \{1, ..., Q\}, && (21) \\ & j = 1, ..., \beta. \end{aligned}$$

We implement this problem using a Metropolis optimization method [14]. We chose this method for its simplicity and the fact that it yields reasonable improvements in $f(\mathbf{P})$ with moderate computing resources. At each iteration, the vector $\mathbf{p}_j$ for every sortie j is perturbed by a random vector within the allowable parameter ranges (20), (21). With these parameters, the vectors $\mu(\mathbf{p}_j, t_s)$, $j = 1, ..., \beta$, are computed by numerically integrating the corresponding model (13) using the Euler method to obtain the solution (19). These vectors are used to evaluate the objective function. For the *open-loop* strategy, all vectors $\mathbf{p}_j$, $j = 1, ..., \beta$, are randomly perturbed simultaneously. To simulate the *closed-loop* strategy, the vector $\mathbf{p}_j$ for a single sortie j is optimized and input to the micro-continuous model, and the resulting distribution of flower visits in each cell i is used as the initial conditions $\mu_{V^i}(\mathbf{p}_j, 0)$ in the optimization of $\mathbf{p}_{j+1}$.

V. SIMULATIONS

We developed micro-continuous and macro-continuous models of a pollination scenario in which a colony of robotic bees is tasked to pollinate 10 rows of rabbiteye blueberry plants with the layout in Fig. 1. Using the macro-continuous model, we optimized the tunable parameters of our system for three objectives: uniform pollination of all plants, different degrees of pollination for cultivars 1 and 2, and uniform pollination in the presence of wind.

The environment is defined as a 140 ft $\times$ 100 ft domain, and the cell length is set to a relatively high value of $l = 4$ ft for fast computation and to reflect the imprecision in the robot localization capabilities. The dimensions in Fig. 1 are set to (x_H, y_H) = (6 ft, 50 ft), $d_H = 16$ ft, and $d_{P,x} = 12$ ft, $d_{P,y} = 6$ ft, the industry standard spacing for this crop [21]. The hive is 4 ft $\times$ 12 ft and each plant is 8 ft $\times$ 6 ft.

We simulated N robots pollinating the field over several sorties, each with a duration of $t_s = 150$ s. The robots begin in a uniform random distribution at the hive. We set $k_{fly} = 0.2$ to define an expected flower visit time of $k_{fly}^{-1} = 5$ s, the average floral handling time at rabbiteye blueberry flowers for species that buzz the flowers for pollen, *Habropoda laboriosa* (2.6 s) and four *Bombus* species (7.4 s) [4]. The parameters were optimized over the following ranges: $v \in [1\ 10]$ ft/s, $D \in [3\ 5]$ ft^2/s (here, $D_{inh} = 3$ ft^2/s), $k_{hov,j} \in [0.05\ 1.25]$ s^{-1} $\forall j$, $r_v \in \{1, ..., 10\}$, $x_B \in [x_H - 50 \quad x_H]$ ft, and $y_B \in [-50\ 150]$ ft. The micro- and macro-continuous models were run with Δt=0.05 s.

A. Uniform Pollination

We set the target number of flower visits per cell to be $N^d_{V^i} = 100\ \forall i \in \mathcal{F}$. For this case, we set $k_{hov,j} \equiv k_{hov}$, $j = 1, ..., Q$. We simulated $N = 1500$ robots executing 3 sorties and used the *open-loop* optimization strategy with

TABLE I

TYPICAL RUNTIMES OF THE MACRO-CONTINUOUS MODEL (τ_{macro}) AND MICRO-CONTINUOUS MODEL (τ_{micro}) ON A 2.60 GHZ PC WITH 3.75 GB RAM VS. ROBOT POPULATION SIZE N

N	1000	2000	3000	5000	7000
τ_{macro} (s)	0.6				
τ_{micro} (s)	11.7	57.4	94.3	255.7	562.7

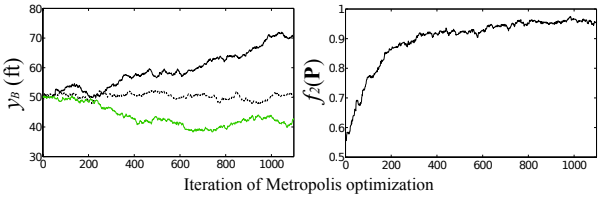

Fig. 2. Values of y_B and $f_2(\mathbf{P})$ for uniform pollination computed at each iteration of the optimization method. In the y_B plot, the dark solid line, dashed line, and light solid line correspond to sorties 1, 2, and 3, respectively.

$f_2(\mathbf{P})$ as the objective function. Table I quantifies the significant computational advantage of using the macro-continuous model rather than micro-continuous model for optimization.

Fig. 2 shows that the stochastic optimization run converges to a set of parameters that produce sufficient pollination over the vast majority of the field, as measured by the metric $f_2 = 0.958$. Table II contains the optimized values of these parameters over all three sorties. As the values of x_B, y_B illustrate, coverage is optimized by relocating the mobile beacon before each sortie in order to deploy the swarm radially outward in different directions. The values of r_v in sorties 1 and 3 contribute to thorough coverage by redirecting the robots eastward as they approach the field boundary.

We performed 30 runs of the micro-continuous model for each sortie with the optimized parameters. Fig. 3 compares the resulting average distributions of flower visits over the field, $N_{V^i}(\mathbf{p}_j, t_s)$, $i = 1, ..., 875$, $j = 1, 2, 3$, to the expected distributions from the macro-continuous model, $\mu_{V^i}(\mathbf{p}_j, t_s)$. The macro-continuous model predicts the general shape of the visit distribution in the micro-continuous model; discrepancies between the models are due to the representation of deterministic motion as fluxes between neighboring cells in the macro-continuous model. The correspondence between the models can be improved by decreasing l. The visit distributions for each sortie in Fig. 3 illustrate the differentiated coverage strategies that are produced by the optimized beacon coordinates (x_B, y_B) and row r_v. The plots in the fourth row, which show the cumulative flower visits over all sorties, confirm that most of the field is adequately pollinated.

B. Nonuniform Pollination

We model a scenario in which we require twice as many flower visits in cultivar 1 as in cultivar 2: $N^d_{V^i} = 100$ $\forall i \in \mathcal{F}_{C1}$ and $N^d_{V^i} = 50$ $\forall i \in \mathcal{F}_{C2}$. We simulated $N = 1500$ robots executing 3 sorties and used the *open-loop* optimization strategy with $f_3(\mathbf{P})$ as the objective function. The optimized values of these parameters over all three sorties are given in Table III.

TABLE II

OPTIMIZED PARAMETERS FOR UNIFORM POLLINATION

Sortie	v (ft/s)	D (ft^2/s)	k_{hov} (s^{-1})	x_B (ft)	y_B (ft)	r_v
1	6.03	3.00	1.02	-13.79	70.11	3
2	5.67	3.24	1.15	-8.77	50.36	10
3	5.59	3.20	1.25	-9.70	42.24	6

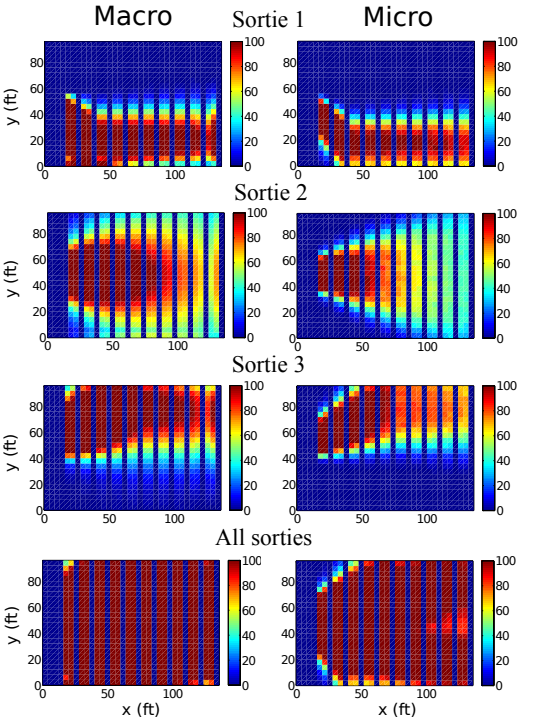

Fig. 3. Distribution of flower visits over 3 sorties in the macro-continuous model (left column) and micro-continuous model (right column) with parameters optimized for uniform pollination. Micro-continuous distributions are averaged over 30 runs. Note that the plots are saturated at $N^d_{V^i} = 100$.

Fig. 4 shows the flower visit distributions from the macro-continuous model and the corresponding micro-continuous model, averaged over 30 runs. As in the uniform case, the optimization produces a sequence of strategies that involve moving the beacon before each sortie to send the swarm in different directions. The optimized $k_{hov,j}$ values are lower for the rows of cultivar 2 than those of cultivar 1 to produce fewer flower visits in cultivar 2. After all sorties, the macro-continuous model closely yields the target visit distribution over the two cultivars. This model slightly overestimates the amount of pollination in the micro-continuous model, which

TABLE III

OPTIMIZED PARAMETERS FOR NONUNIFORM POLLINATION

Sortie	v (ft/s)	D (ft^2/s)	x_B (ft)	y_B (ft)	r_v	$k_{hov,j}$ (s^{-1}), $j=$ 1	2
1	7.11	4.44	-13.91	68.13	6	0.37	0.29
2	8.72	3.19	-9.89	29.25	4	0.45	0.32
3	3.83	3.70	-5.51	48.71	6	0.29	0.21

Sortie	$k_{hov,j}$ (s^{-1}), $j=$ 3	4	5	6	7	8	9	10
1	0.67	0.32	0.85	0.48	1.19	0.79	1.18	0.68
2	0.82	0.36	0.76	0.29	0.68	0.28	1.04	0.44
3	0.61	0.46	1.07	0.65	1.21	0.61	1.21	0.65

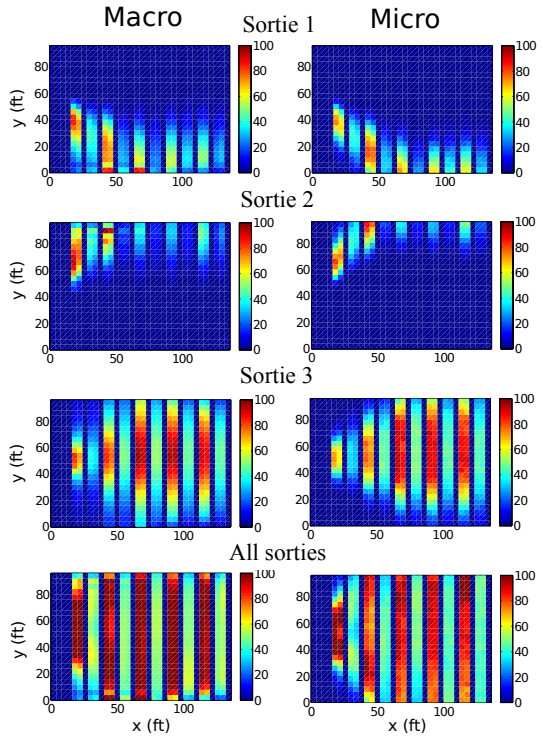

Fig. 4. Distribution of flower visits over 3 sorties in the macro- and micro-continuous models with parameters optimized for nonuniform pollination.

approximately displays the desired ratio in the degree of pollination between the two cultivars.

C. Uniform Pollination in the Presence of Wind

We set $N^d_{V^i} = 100\ \forall i \in \mathcal{F}$ to achieve uniform pollination. In this scenario, a light southeast wind with velocity $\mathbf{v}_w = [1\ \ -2]^T$ ft/s blows over the entire field. This vector is added to the robot velocity field (1) in the micro-continuous model. The *closed-loop* strategy was used in order to adapt the robot behaviors to the wind over 5 sorties. We simulated the cycle of swarm deployment and parameter optimization/transmission at the hive with $N = 1500$ and $N = 3000$ robots. We set $k_{hov,j} \equiv k_{hov}$, $j = 1, ..., Q$ and used $f_1(\mathbf{P})$ as the objective function. The optimized parameter values for $N = 3000$ are given in Table IV.

TABLE IV

OPTIMIZED PARAMETERS FOR UNIFORM POLLINATION WITH WIND

Sortie	v (ft/s)	D (ft^2/s)	k_{hov} (s^{-1})	x_B (ft)	y_B (ft)	r_v
1	5.46	3.00	1.25	-5.14	49.80	10
2	5.48	3.00	1.25	-3.67	48.43	10
3	5.29	3.00	1.24	-0.72	49.24	10
4	4.40	3.00	1.25	-0.42	49.98	10
5	5.72	3.67	1.25	0.93	49.03	6

Fig. 5 shows the flower visit distributions from the macro-continuous model and single runs of the micro-continuous model. The flower visit distribution in the micro-continuous model always extends farther southeast than predicted by the macro-continuous model due to the drift added to the robots by the wind. For a swarm of 1500 robots, the repeated optimization of the parameters for more thorough pollination of the upper half of the field still left significant regions unvisited after 5 sorties, as shown by the micro-continuous visit distribution in the last row of Fig. 5. The other plots illustrate that deploying twice as many robots yields a substantial increase in the extent of sufficient pollination. The optimized values of x_B, y_B indicate that the beacon is moved closer to the hive for sorties $3-5$, which widens the swarm dispersal angle θ in Fig. 1 in an effort to cover the unpollinated upper and lower left corners of the field.

VI. CONCLUSIONS AND FUTURE WORK

We have described a methodology for optimizing robot control policies for a desired global behavior in a swarm whose members may be arbitrarily distributed in space and execute stochastic task transitions. The parameters of the robot motion and task switching are optimized using an abstraction of the physical system. This macroscopic model describes the expected populations of different swarm elements over a discretized domain; its accuracy increases with the grid resolution, which is constrained by the robot localization capabilities if the optimization incorporates robot feedback. We demonstrate the ability of our methodology to approximately produce, using a relatively coarse discretization, target distributions of pollination over a field by robotic insects, including in the presence of an unknown wind.

We intend to investigate ways to increase the accuracy of the macro-continuous model for a given grid resolution. For the *closed-loop* strategy, the optimization method can include an estimation of the unknown wind velocity, based on the robot data on flower visits, to incorporate into k^v_{ij}. We are also interested in applying our methodology to models

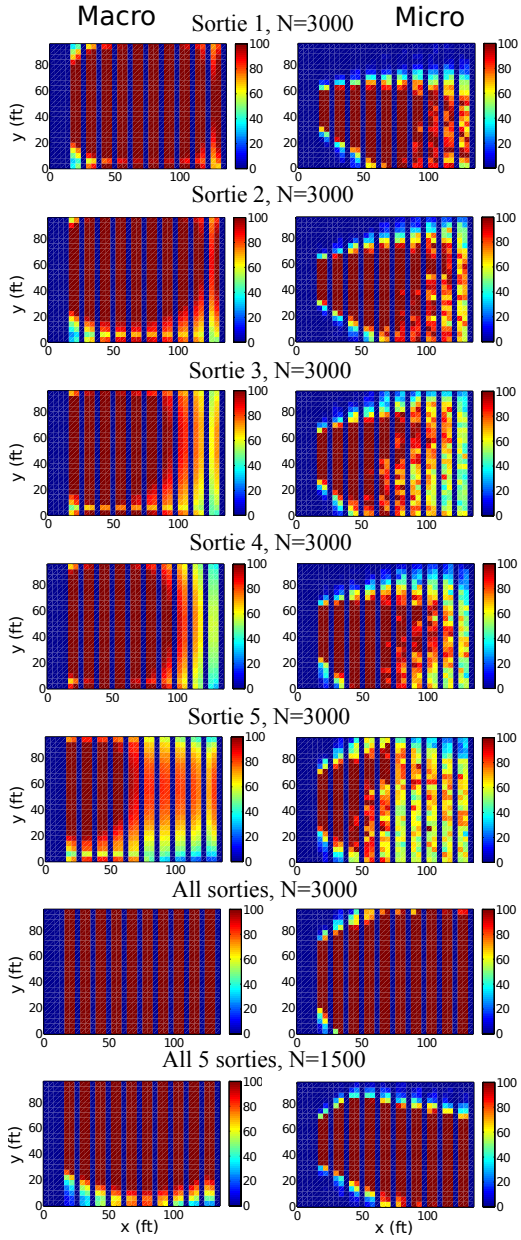

Fig. 5. Distribution of flower visits over 5 sorties in the macro-continuous model and single runs of the micro-continuous model with parameters optimized for uniform pollination in the presence of wind. Results are shown for robot population sizes of $N = 3000$ and $N = 1500$.

for our system that include inter-robot communication, robot failures, and the asynchronous return of robots to the hive triggered by low remaining power levels.

VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge support from NSF Award CCF-0926148.

REFERENCES

[1] *Robobees Project*. http://robobees.seas.harvard.edu/, 2011.

[2] S. Berman, Á. Halász, M. A. Hsieh, and V. Kumar. Optimized stochastic policies for task allocation in swarms of robots. *IEEE Trans. Robotics*, 25(4):927–937, Aug. 2009.

[3] S. Berman, V. Kumar, and R. Nagpal. Design of control policies for spatially inhomogeneous robot swarms with application to commercial pollination. In *Int'l. Conf. Robotics and Automation (ICRA)*, 2011.

[4] J. H. Cane and J. A. Payne. Foraging ecology of the bee Habropoda laboriosa (Hymenoptera: Anthophoridae), an oligolege of blueberries (Ericaceae: Vaccinium) in the southeastern United States. *Ann. Entomol. Soc. Am.*, 81(3):419–427, 1988.

[5] N. Correll. Parameter estimation and optimal control of swarm-robotic systems: A case study in distributed task allocation. In *Int'l. Conf. Robotics and Automation (ICRA)*, pages 3302–3307, 2008.

[6] K. S. Delaplane and D. F. Mayer. *Crop Pollination by Bees*. CABI Publishing, New York, NY, 2000.

[7] W. C. Evans, G. Mermoud, and A. Martinoli. Comparing and modeling distributed control strategies for miniature self-assembling robots. In *Int'l. Conf. Robotics and Automation (ICRA)*, pages 1438–1445, 2010.

[8] J. B. Free. *Insect Pollination Of Crops*. Academic Press, London, UK, 2nd edition, 1993.

[9] C. Gardiner. *Stochastic Methods: A Handbook for the Natural and Social Sciences*. Springer, 4th edition, 2009.

[10] Dan Gillespie. Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, 58:35–55, 2007.

[11] 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.

[12] M. Karpelson, J. P. Whitney, G.-Y. Wei, and R. J. Wood. Energetics of flapping-wing robotic insects: towards autonomous hovering flight. In *Int'l. Conf. Intell. Robots and Syst. (IROS)*, pages 1630–1637, 2010.

[13] Eric Klavins, Samuel Burden, and Nils Napp. Optimal rules for programmed stochastic self-assembly. *Robotics: Science and Systems II*, pages 9–16, 2007.

[14] D. P. Landau and K. Binder. *A Guide to Monte-Carlo Simulations in Statistical Physics*. Cambridge Univ. Press, 2000.

[15] R. J. Leveque. *Finite-Volume Methods for Hyperbolic Problems*. Cambridge Univ. Press, 2004.

[16] W. Liu and A. F. T. Winfield. Modeling and optimization of adaptive foraging in swarm robotic systems. *Int'l. J. of Robotics Research*, 29(14):1743–1760, 2010.

[17] L. Matthey, S. Berman, and V. Kumar. Stochastic strategies for a swarm robotic assembly system. In *Int'l. Conf. Robotics and Automation (ICRA)*, pages 1953–1958, 2009.

[18] N. Michael, J. Fink, S. Loizou, and V. Kumar. Architecture, abstractions, and algorithms for controlling large teams of robots: Experimental testbed and results. *Int'l. Symposium of Robotics Research (ISRR)*, 2007.

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

[20] L. Odhner and H. Asada. Stochastic recruitment control of large ensemble systems with limited feedback. *ASME J. Dyn. Sys. Meas. Control*, 132(4):041008–1–041008–9, 2010.

[21] A. Powell, W. A. Dozier Jr., and D. G. Himelrick. Commercial blueberry production guide for Alabama. 2002. Alabama Cooperative Extension System, ANR-904.

[22] A. Prorok, N. Correll, and A. Martinoli. Multi-level spatial modeling for stochastic distributed robotic systems. *Int'l. J. of Robotics Research*, 30(5):574–589, 2011.

[23] R. J. Wood. The first takeoff of a biologically inspired at-scale robotic insect. *IEEE Trans. Robotics*, 24(2):341–347, 2008.