Algorithms for the Analysis and Synthesis of a Bio-inspired Swarm Robotic System

Spring Berman[1], Ádám Halász[1], Vijay Kumar[1], and Stephen Pratt[2]

[1] University of Pennsylvania, Philadelphia PA 19104, USA
{spring,halasz,kumar}@grasp.upenn.edu
[2] Arizona State University, Tempe AZ 85287, USA
Stephen.Pratt@asu.edu

Abstract. We present a methodology for characterizing, analyzing, and synthesizing swarm behaviors using both a macroscopic continuous model that represents a swarm as a continuum and a macroscopic discrete model that enumerates individual agents. Our methodology is applied to a dynamical model of ant house hunting, a decentralized process in which a colony attempts to emigrate to the best site among several alternatives. The model is hybrid because the colony switches between different sets of behaviors, or modes, during this process. Using the model in [1], we investigate the relation of site population growth to initial system state with an algorithm called Multi-Affine Reachability analysis using Conical Overapproximations (MARCO) [2]. We then derive a microscopic hybrid dynamical model of an agent that respects the specifications of the global behavior at the continuous level. Our multi-level simulations demonstrate that we have produced a rigorously correct microscopic model from the macroscopic descriptions.

Keywords: multiscale modeling, synthesis, abstractions of swarms, reachability analysis, stochastic simulation, insect house hunting.

1 Introduction

Coordinated multi-agent systems have yielded robust, efficient, and cost-effective solutions to diverse objectives, such as the establishment of a mobile sensor network for environmental monitoring, surveillance, or reconnaissance; object manipulation and transportation; and search-and-rescue tasks. One multi-agent paradigm is a swarm robotic system, which consists of many anonymous agents that operate autonomously under decentralized control laws. Although each agent follows relatively simple rules, the group can collectively achieve complex tasks at the macroscopic level. In this sense, robot swarm systems can draw inspiration from the self-organized processes of natural aggregations such as social insect colonies [3], which accomplish global objectives such as nest construction, foraging, brood sorting, and colony relocation through local interactions, both among individuals and between individuals and their environment. In a robotics

E. Şahin et al. (Eds.): Swarm Robotics Ws, LNCS 4433, pp. 56–70, 2007.

context, the simplicity and identical nature of swarm agents offer the advantages of system robustness and control scalability with population size.

Our goal in this paper is to establish a general methodology to solve the so-called *inverse problem*: the design of individual behaviors to achieve a desired macroscopic behavior for the group. This work is related in spirit to the work of [4], which presents a systematic approach to translate group behaviors, modeled as vector fields on a low-dimensional abstract manifold, into agent behaviors in a high-dimensional manifold derived from copies of an agent's state space. As in recent work on modeling and analyzing swarm robotic systems [5] [6] [7], we employ a multi-level representation of swarm activity. At the highest level, we consider a *macro-continuous* model, also called the Rate Equation model [8], characterized by differential equations in which the state variables represent population fractions engaged in different tasks or roles. We distinguish the *macro-discrete* level, which models a discrete number of agents in each task according to the Stochastic Master Equation [9], as an intermediate level. This level permits behaviors synthesized at the highest level to be translated into difference equations involving integers, in effect representing the system as a finite automaton. At the bottom of the hierarchy, the *microscopic* level [8] models agents in a physical setting, incorporating the geometry and dynamics of individual agents and possibly modeling heterogeneity.

Several types of distributed robot systems have been modeled by translating an individual robot controller into a description of collective behavior. Collaborative stick-pulling [5] and object clustering [6] have been modeled with Probabilistic Finite State Machines, whose states represent both (a) the possible behaviors of a single agent at the microscopic level, and (b) the average number of agents in each task at a certain time step at the macro-continuous level. The robots obey the semi-Markovian property: their state transitions depend only on their present state and the amount of time they have occupied the state. Adaptive robots that change their task based on a history of local observations have been modeled in a multi-foraging scenario [7]. In this work, the macro-continuous and macro-discrete levels are derived from a microscopic model that abstracts away physical robot behaviors. In all of the systems, it is assumed that robots and their stimuli are uniformly spatially distributed. The state transition rates in the macro-continuous model are computed from physical robot parameters, sensor measurements, and geometrical considerations. The macro-continuous models are validated by comparing steady-state variables and other quantities of interest to the results of embodied simulations and experiments.

In addition to the bottom-up design methodology just described, a top-down approach has been used to synthesize agent controllers [10]. An algorithm is first designed assuming that agents have global information, which is then replaced with local information that is exchanged among agents. In an application to the multi-foraging scenario [7], the probability of a state transition is generated through the gradient descent of an objective function with a minimum at the robot state distribution that matches the task type distribution.

In the methods just discussed, the main challenge is to derive an appropriate mathematical form for the task state transition rates [8]. In contrast, the top-down design approach that we present assumes that these rates are known beforehand. To investigate the effect of changing them, we do not need to simulate the system under many different conditions; instead, our macroscopic analysis technique allows us to determine the global influence of ranges of such parameters. In addition, we provide a framework for synthesizing a desired system outcome and then translating the macroscopic behaviors into individual agent behaviors. This technique does not require progressive model decentralization and calibration of the fully distributed system, as does the top-down approach in [10]. Finally, our macro-discrete level can capture phenomena that occur at the microscopic level but are lost at the macro-continuous level. This is because the stochastic formulation of a system has a more legitimate physical basis than the deterministic formulation [9]. Examples of such phenomena include state fluctuations in relatively small populations, potentially leading to stochastic transitions between equilibria of multi-stable systems [11].

We apply our methodology to a model inspired by the work of [1], which studies the process by which a colony of *Temnothorax albipennis* ants chooses a new home from several sites and emigrates through quorum-dependent recruitment mechanisms. The quorum dependency creates a hybrid system in which the ants switch tasks, which can be thought of as sets of controllers, based on their surroundings. The quorum sensing mechanism is key to the collective decision-making process of nest site selection. The authors present models at the two macroscopic levels. However, because they were not interested in models of individual ants and their dynamics, the microscopic-level modeling is absent.

From a robotics perspective, an analogy can be drawn between the ants and robotic agents with limited communication that must distribute themselves or transport objects optimally among several locations. We are concerned with three interesting questions on the biological phenomena and their implications for robotics. (1) Why do the ants behave as they do and is their behavior optimal in any sense? (2) Can we prove that this behavior leads to successful migration to the best nest? (3) Can this behavior be realized on robotic systems? Our paper addresses the second and third questions. We answer (2) using a reachability analysis technique that permits us to explore all possible states reached by a macro-continuous level model. We answer (3) by deriving a methodology that allows macro-continuous level abstract behaviors (as in [4]) to be realized at the macro-discrete level and then at the microscopic level.

2 Methodology

We consider a population of N agents moving in the continuous state space $X_a \subset \mathbb{R}^2$. At any given time, an agent's actions are determined by one of a set L_a of l_a controllers or behaviors. We can describe the agent as a hybrid automaton, $H_a = \{X_a, L_a\}$, to indicate that its activity is governed by both continuous and discrete dynamics. Figure 1a shows how this high-dimensional microscopic level can be mapped to lower-dimensional representations, the macro-discrete and macro-continuous levels, through the abstractions $\mathcal{F}_d$ and $\mathcal{F}_c$, respectively.

Representing the swarm as a continuous quantity, the macro-continuous level models its dynamics with a set of differential equations whose variables, x_i $(i = 1, ..., b)$, are the population fractions associated with different tasks or roles. Each agent mode $l \in L_a$ corresponds to one of these tasks, and possibly to a subdivision of activity within a task. It is assumed that the population is conserved, so one variable may be removed through the conservation constraint. The variables therefore comprise a continuous state space $X_p \subset \mathbb{R}^{b-1}$. If the model is a hybrid system, then the state space is divided into a set L_p of l_p regions, called population modes, each of which is associated with different continuous dynamics. The system may then be described by a hybrid automaton $H_p = \{X_p, L_p\}$. The macro-discrete level, which considers a swarm as a collection of discrete agents rather than a continuum, maintains a count of the number of agents in each of the b tasks or roles.

As Figure 1b shows, our methodology for designing a swarm system and analyzing its behavior relies on all three of these levels of abstraction. We shall illustrate our methodology with the concept of an emigrating ant colony whose rules of behavior, either known (as in biological systems) or designable (as in artificial swarms), are within our control.

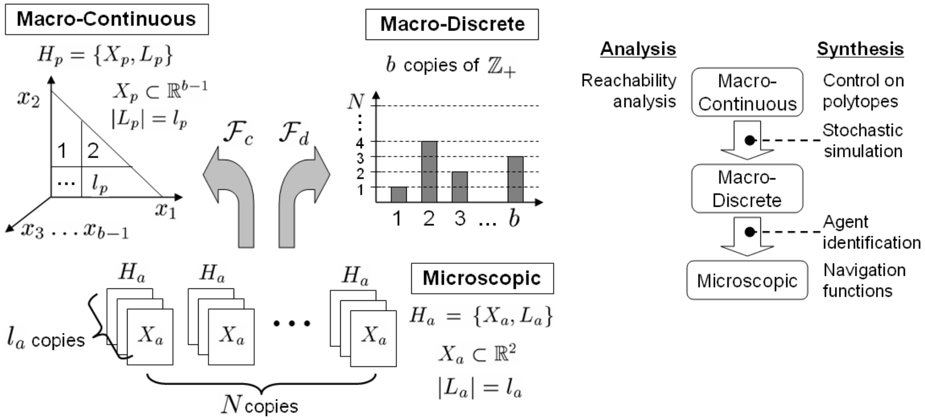


Fig. 1. (a) Levels of abstraction of a swarm; (b) Analysis and synthesis methodologies

The macro-continuous level is used to define and plan the execution of the general task that the system should achieve. In nature, a signature of self-organizing systems is multi-stability, with the most adequate stable states selected according to their fitness [12]. This trait lends robustness to the system under perturbations. In our engineered house-hunting scenario, however, we may want to control the system so that there is always one outcome: the emigration of the entire colony to the optimal nest among several available sites. In addition, let us suppose that we want the emigration to allow at most a fraction of the colony, say 25%, to be separated from the rest for no more than T time units.

The macro-continuous level analysis checks whether or not the continuous model satisfies the requirements. The first condition can be verified by using steady-state analysis to ensure that the model has a single stable equilibrium that corresponds to the entire colony's settlement in the best site. The traditional approach to checking the second condition, or to identifying a range of parameter values that produces a desired result, is to solve the continuous model for many different initial states and parameter sets. This verification can be done more efficiently with *reachability analysis*, which determines the set of states that are attainable from an initial set A. If set B consists of the states in which over 25% of the colony is separated from the rest, then the analysis can show whether (1) A ever reaches B and, if so, whether (2) the system remains there for longer than time T. Problem (1) is a standard reachability question that can be investigated by overapproximating the reachable set on a discrete abstraction of the system or on the state space directly. Problem (2) can be converted into (1) by adding a clock s for which $\dot{s} = 1$ if the system is in B and $\dot{s} = 0$ otherwise, and seeing whether the augmented system reaches the set $s > T$. Similarly, the system behavior over a parameter range can be analyzed by adding the parameter p as a state with $\dot{p} = 0$ and including an interval over p in set A. The macro-continuous model may be solved with the parameters that are chosen from this analysis to ensure that they produce the desired system evolution.

If reachability analysis reveals that the system exhibits undesirable behavior, then control terms can be added to the macro-continuous model to meet the requirements. [13] presents a method of defining feedback control laws on a piecewise-linear hybrid system. Control inputs are defined at the vertices of a polytope state-space region that corresponds to a mode, and a convex combination of these inputs is used to drive states inside the polytope to the next desired mode.

The macro-discrete level connects the macro-continuous level to the microscopic level, which is needed for the ultimate implementation. This level is still a macroscopic model that abstracts away agent identities; however, now we consider an integer number of agents. We simulate transitions between tasks by incrementing and decrementing the number of agents in each task. To synthesize this level, we apply a simulation algorithm from [9], which has been used in the mathematically similar problem of replacing a differential equation description of chemical kinetics with individual molecular reactions. The algorithm generates a sequence of transitions and their times according to a probability density function that is rigorously derived from the known physical principles that govern the underlying chemical processes. In our case, the rules producing the transitions may be stochastic, intrinsically and/or by design, or deterministic. Transition times in the house-hunting model are governed by a Poisson distribution. As $N \to \infty$, the Poisson transition probabilities per unit time become transition rates, and the macro-discrete level simulation approaches the macro-continuous level solution. Transitions have a deterministic component if they are delayed by the time an agent takes to perform an action necessary to change its state, such as navigation.

If we wish to construct a robot swarm that behaves similarly to the ant colony, we need to prescribe the behavior of each agent in all situations it may encounter. This occurs at the microscopic level, where agent identities and spatial considerations become important. At this level, travel between two sites is implemented using navigation functions [14], which can be defined on environments of a certain topological class to guide an agent to a goal while steering it away from obstacles. The resulting mean travel times are used as time delays in the macro-discrete level. For the microscopic level to be abstracted to the macro-discrete level, state transitions should not depend on the previous history of the agent (the Markov property), and spatial information must be either discarded or converted into substates associated with regions in the physical space.

We note that aside from its specification of navigation controllers, our microscopic model is still a coarse-grained representation [8] since it abstracts away ant behaviors such as quorum estimation, recruiter-recruitee communication, and avoidance of collisions with other ants. Thus, the model still requires more detail in order to constitute an executable robot controller. We point out that the quorum dependency does not pose a theoretical impediment to synthesizing such a controller. In our model, only the ants that visit a nest know whether it has attained a quorum population. From the perspective of transition dynamics, an ant that has perceived a quorum is in a different state than an ant that has not, but the two ants are otherwise identical. Therefore, the quorum condition does not violate the Markov property of the model.

3 Macro-continuous Model

Our model of ant house hunting behavior is an extension of the one presented in [1], which was constructed from experimental observations of *Temnothorax albipennis*. Although we try to reflect ant behavior as accurately as possible, our main goal is not to create a new description of ant house hunting, which has already been modeled in considerable detail [15]. Instead, our objective is to make the original model in [1] realizable on the microscopic level, with the ultimate purpose of synthesizing robot controllers that will produce ant-like activity.

The model consists of a set of coupled delay differential equations whose state variables represent population fractions that are physically located at the home nest or one of the M potential home sites. The time delays are averages of navigation times between sites from the microscopic simulation described in section 5.2. Each ant has knowledge of at most two sites, one of which is its home. A colony of N ants is divided into a fraction p of active ants and a remainder of passive ants. The active ant fraction is divided among the following state variables. Naive ants, Y_i, reside at site i, which they consider their home; they leave this site to search for a new nest. Assessing ants, Z_{ij}, regard site i as their home and are evaluating site j as a potential new home. Recruiting ants, $Y_{ij,n}$, are located at site $n \in \{i, j\}$ and leave to bring other ants from i to j. The method of recruitment of $Y_{ij,j}$ ants depends on the population fraction located at site j, P_j. If P_j has not reached a quorum Q, then $Y_{ij,j}$ ants still consider site i to be their home, and they limit themselves to using tandem runs to lead fellow active ants in one of b_{tand} states, Y_i and $Z_{k,i}$ ($k \neq i, j$), to assess site j. If $P_j \geq Q$, then site j becomes their home and they use transports to carry the passive ants at site i, B_i, to site j. $Y_{ij,i}$ ants always recruit via transports. When $Y_{ij,n}$ ants realize that there are no B_i ants left to transport, they "forget" site i and become naive ants at site j, Y_j.

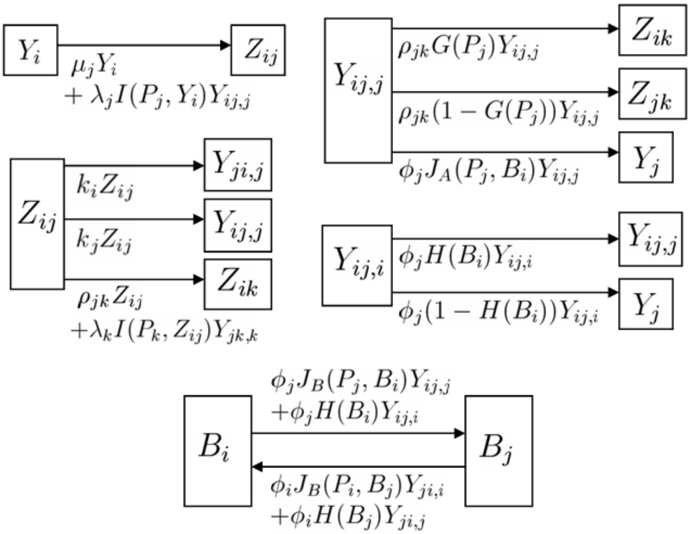
The rates in the model were experimentally derived [16]. Naive ants discover site i at per capita rate μ_i. Assessors become recruiters to site i at per capita rate k_i, which is directly related to the quality of the site. λ_i and ϕ_i are the per capita rates at which recruiters perform tandem runs and transports to site i, respectively. ρ_{ij} is the per capita rate at which assessors and recruiters at site i encounter site j and switch their allegiance by becoming assessors of that site.

The model is defined by equations (1)-(5). For a variable X, $X = X(t)$ and $X[\tau_{ij}] = X(t - \tau_{ij})$. The time delay τ_{ij} represents the time taken to travel from site i to site j; $\tau_{ji+ij} = \tau_{ji} + \tau_{ij}$. If i and j are in bold, unitalicized font, then the trip is a tandem run. To illustrate the state transitions, the flowchart in Figure 2 diagrams the model with all time delays set to zero.

$$\begin{aligned}
\dot{Y}_i &= \sum\nolimits_{\substack{j=0\\ j\neq i}}^{M} [\phi_i J_A(P_i[\tau_{ij+ji}], B_j[\tau_{ji}]) Y_{ji,i}[\tau_{ij+ji}] + \phi_i (1 - H(B_j[\tau_{ji}])) Y_{ji,j}] \\
&\quad - \sum\nolimits_{\substack{j=0\\ j\neq i}}^{M} [\lambda_j I(P_j[\tau_{ji}], Y_i) Y_{ij,j}[\tau_{ji}] + \mu_j Y_i] \qquad (1)\\
\dot{Z}_{ij} &= \mu_j Y_i[\tau_{ij}] - (k_i + k_j) Z_{ij} + \sum\nolimits_{\substack{k=0\\ k\neq i,j}}^{M} [\rho_{kj} Z_{ik}[\tau_{kj}] - \rho_{jk} Z_{ij}] \\
&\quad + \sum\nolimits_{\substack{k=0\\ k\neq i,j}}^{M} [\rho_{ij} (1 - G(P_i[\tau_{ij}])) Y_{ki,i}[\tau_{ij}] + \rho_{kj} G(P_k[\tau_{kj}]) Y_{ik,k}[\tau_{kj}]] \\
&\quad + \sum\nolimits_{\substack{k=0\\ k\neq i,j}}^{M} [\lambda_j I(P_j[\tau_{jk+\mathbf{kj}}], Z_{ik}[\tau_{\mathbf{kj}}]) Y_{kj,j}[\tau_{jk+\mathbf{kj}}] - \lambda_k I(P_k[\tau_{kj}], Z_{ij}) Y_{jk,k}[\tau_{kj}]] \\
&\quad + \lambda_j I(P_j[\tau_{ji+\mathbf{ij}}], Y_i[\tau_{\mathbf{ij}}]) Y_{ij,j}[\tau_{ji+\mathbf{ij}}] \qquad (2)\\
\dot{Y}_{ij,i} &= k_j Z_{ji} - \phi_j Y_{ij,i} \qquad (3)\\
\dot{Y}_{ij,j} &= k_j Z_{ij} + b_{tand}[-\lambda_j G(P_j) Y_{ij,j} + \lambda_j G(P_j[\tau_{ji+\mathbf{ij}}]) Y_{ij,j}[\tau_{ji+\mathbf{ij}}]] \\
&\quad - \phi_j (1 - G(P_j)) Y_{ij,j} + \phi_j J_B(P_j[\tau_{ji+ij}], B_i[\tau_{ij}]) Y_{ij,j}[\tau_{ji+ij}] \\
&\quad + \phi_j H(B_i[\tau_{ij}]) Y_{ij,i}[\tau_{ij}] - \sum\nolimits_{\substack{k=0\\ k\neq i,j}}^{M} \rho_{jk} Y_{ij,j} \qquad (4)\\
\dot{B}_i &= \sum\nolimits_{\substack{j=0\\ j\neq i}}^{M} [\phi_i J_B(P_i[\tau_{ij+ji}], B_j[\tau_{ji}]) Y_{ji,i}[\tau_{ij+ji}] + \phi_i H(B_j[\tau_{ji}]) Y_{ji,j}[\tau_{ji}]] \\
&\quad - \sum\nolimits_{\substack{j=0\\ j\neq i}}^{M} [\phi_j J_B(P_j[\tau_{ji}], B_i) Y_{ij,j}[\tau_{ji}] + \phi_j H(B_i) Y_{ij,i}] \qquad (5)
\end{aligned}$$

$$P_j = Y_j + B_j + \sum\nolimits_{\substack{i=0\\ i\neq j}}^{M} [Y_{ij,j} + Y_{ji,j} + Z_{ij}]$$

$$G(P) = 1 \quad \text{if } P < Q; \;\; 0 \text{ otherwise}$$
$$H(B) = 1 \quad \text{if } B > 0; \;\; 0 \text{ otherwise}$$
$$I(P, X) = 1 \quad \text{if } P < Q \text{ and } X > 0; \;\; 0 \text{ otherwise}$$
$$J_A(P, B) = 1 \quad \text{if } P \geq Q \text{ and } B = 0; \;\; 0 \text{ otherwise}$$
$$J_B(P, B) = 1 \quad \text{if } P \geq Q \text{ and } B > 0; \;\; 0 \text{ otherwise}$$



Y_i
Z_{ij}
$\mu_j Y_i$
$+ \lambda_j I(P_j, Y_i) Y_{ij,j}$
$Y_{ij,j}$
Z_{ik}
$\rho_{jk} G(P_j) Y_{ij,j}$
Z_{jk}
$\rho_{jk}(1 - G(P_j)) Y_{ij,j}$
Y_j
$\phi_j J_A(P_j, B_i) Y_{ij,j}$
Z_{ij}
$Y_{ji,j}$
$k_i Z_{ij}$
$Y_{ij,j}$
$k_j Z_{ij}$
Z_{ik}
$\rho_{jk} Z_{ij}$
$+\lambda_k I(P_k, Z_{ij}) Y_{jk,k}$
$Y_{ij,i}$
$Y_{ij,j}$
$\phi_j H(B_i) Y_{ij,i}$
Y_j
$\phi_j(1 - H(B_i)) Y_{ij,i}$
$\phi_j J_B(P_j, B_i) Y_{ij,j}$
$+\phi_j H(B_i) Y_{ij,i}$
B_i
B_j
$\phi_i J_B(P_i, B_j) Y_{ji,i}$
$+\phi_i H(B_j) Y_{ji,j}$

Fig. 2. Flowchart for ant house hunting dynamics without time delays

4 Reachability Analysis

4.1 Algorithm

The Marco reachability algorithm [2] is written in Matlab and uses the Multi-Parametric Toolbox (MPT) for polyhedral operations. The algorithm, which generates reachable sets of a hybrid system H_p in the state space X_p, was developed to compute more precise and accurate reachable sets than an existing method [17], particularly for systems with multi-affine dynamics. The algorithm begins by initializing a list of reachable modes with the modes that contain the initial set. These modes are identified as members of generation 0. The portion of the initial set that each mode contains is considered its first "footprint." For each mode, a truncated cone is defined as the convex hull of the origin and the state derivatives at the mode vertices. The cone is scaled, added to the mode's footprint via a Minkowski sum, and bounded by the mode facets. The resulting set of states represents an overapproximation of the paths that all points in the footprint can traverse within the mode. Next, each neighboring mode with a facet that intersects this reached set is added to the list of reachable modes, and the intersection is designated as the footprint of that mode. These modes are identified as members of the next generation.

The algorithm repeats the reachable set overapproximation and footprint identification for modes in each consecutive generation. If a mode has multiple footprints, the union of their conical reached sets is the total reachable set within the mode. The algorithm terminates when the reachable set for each mode in a generation is a subset of the set already computed for these modes. It may also terminate if there are no new modes in the current generation, which occurs when the reachable set hits the boundary of X_p.

4.2 Application to the House-Hunting Model

We applied our algorithm to the macro-continuous model in [1] to identify sets of initial conditions that guarantee that a particular nest site reaches a quorum before the other site. This model is a special case of the model (1)–(5) and does not include time delays due to navigation. There are three nest sites, labeled 0, 1, and 2. Site 0 is the home nest, which has been destroyed and therefore does not attract recruitment. P_i is equal to the number of recruiters to site i, $Y_{0i,i}$. There are five active ant state variables $(Y_0, Y_{01,1}, Y_{02,2}, Z_{01}, Z_{02})$, which are decoupled from the three passive ant state variables (B_0, B_1, B_2). Thus, after eliminating Y_0 through the active ant conservation constraint, the full analysis region is the four-dimensional state space $\{Y_{01,1}, Y_{02,2}, Z_{01}, Z_{02} \geq 0,\ Y_{01,1} + Y_{02,2} + Z_{01} + Z_{02} \leq p\}$.

The state space is divided into modes by the hyperplanes $P_1 = Q$ and $P_2 = Q$, the quorum switches. The analysis focuses on a portion of the mode that is bounded by these hyperplanes. The analysis region is set to $Y_{01,1}, Y_{02,2} \in [0, 0.0481]$, $Z_{01}, Z_{02} \in [0, 0.0721]$ and divided into modes of dimension $0.0120 \times 0.0120 \times 0.0144 \times 0.0144$ for refinement of the reachable set. Initial box A is defined as $Y_{01,1} \in [0.0337, 0.0385]$, $Y_{02,2} \in [0, 0.00481]$, $Z_{01}, Z_{02} \in [0.0288, 0.0337]$; initial box B is $Y_{01,1} \in [0, 0.00481]$, $Y_{02,2} \in [0.0240, 0.0288]$, $Z_{01}, Z_{02} \in [0.0288, 0.0337]$.

In Figure 3, the unions of gray polygons are two-dimensional projections of the reachable set from each initial box. The computation took 33.5 minutes and consisted of 8 generations for box A and 22.3 minutes, 9 generations for box B. Each four-dimensional box has 16 vertices, which are projected onto the $Y_{01,1} - Y_{02,2}$ plane. The black lines are the solutions of the continuous model starting at these vertices. As shown by comparison with these solutions, both reachable sets correctly predict the first site to achieve a quorum of 0.0481. The reachability results show that all system trajectories starting inside box A and box B will first cross the quorum for site 1 and site 2, respectively. The algorithm guarantees this without computing any of the actual trajectories.

5 Simulation

5.1 Algorithms

Macro-Continuous Level. The system of equations (1)–(5) can be numerically integrated using standard techniques such as the Runge-Kutta method.

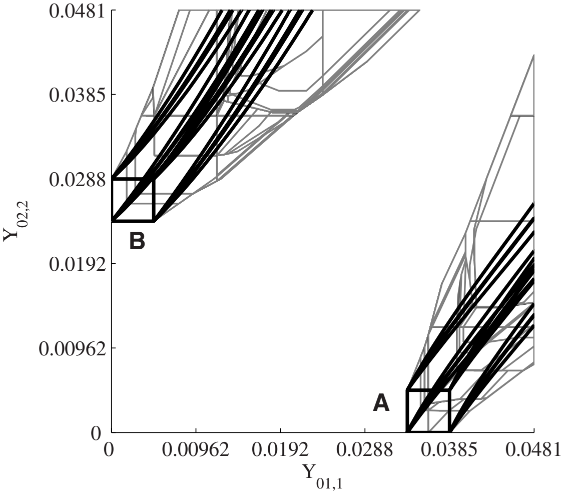


Fig. 3. Two-dimensional projection of reachable sets; $p = 0.25$, $Q = 0.0481$, $\mu_1 = \mu_2 = 0.013$, $\lambda_1 = \lambda_2 = 0.033$, $\rho_{12} = 0.004$, $k_1 = 0.019$, $k_2 = 0.020$ (values are from [1], [16])

Macro-Discrete Level. Gillespie's Direct Method [9] was used to perform a stochastic simulation of the system that is represented deterministically by the macro-continuous model. This method was originally devised to numerically calculate the time evolution of chemical reactions. Like a system of reactions, the macro-continuous model (1)–(5) is described by a set of coupled differential equations. Consider the model without time delays in Figure 2. Each of the S possible transformations of x_i into x_j, the population fractions in task states i and j, respectively, is governed by a term of the form $k_{s0} f(P_q, x_r) x_m$. The fraction x_i is analogous to the molecular concentration of a chemical species, and k_{s0} is analogous to a deterministic reaction-rate constant. When $m \neq i$, m is a recruiter state. $f(P_q, x_r) = 1$ when the transformation is not governed by a switch. Otherwise, it is 0 or 1 depending on P_q, the population fraction at the site containing ants in state m, and/or x_r, the fraction of ants in a recruitee state r. We can remove the switch dependence on x_r, since an ant that decides to recruit does not immediately know about the availability of recruitees. This dependence is replaced with a deterministic state transition of the recruiter based on the presence of recruitees once the recruiter reaches their site.

To construct a stochastic formulation of the system, we convert the macro-continuous model into a set of unidirectional "reactions" with one "reactant ant" X_i and one "product ant" X_j. These reactions describe individual state transitions. The transition is enabled only when $f(P_q) = 1$. Like a chemical reaction, each transition is characterized by a parameter c_s such that $c_s dt$ is the average probability that a particular ant in state i will undergo transition s in the next time interval dt. Since each transition has only one "reactant ant," $c_s = k_s$ [9]. The parameter c_s for transition s is computed by setting the original term

$k_{s0}x_m = c_{s0}x_m$ equal to a new term $c_s x_i$. In this way, we generate S transitions $X_i \rightarrow X_j$ with parameter $c_s = c_{s0}x_m/x_i$.

The propensity a_s is defined such that $a_s dt$ is the probability that transition s will occur in the next time interval dt. It is the product of c_s with h_s, the current number of distinct "reactant ant" combinations that can undergo the transition. Because each transition has only one "reactant ant" X_i, h_s is the number of ants in state i, $n_i = x_i N$ [9]. Thus, $a_s = c_s n_i = c_{s0}n_m$. The propensity a_s is zero if $n_m = 0$ or if transition s is disabled by a switch term $f(P_q)$.

The Direct Method is implemented in the following way. First, the number of ants in each state is initialized in a counter and the S propensities are calculated. The next state transition is selected according to a uniform probability distribution over the propensities, and the time until its occurrence, $\Delta\tau$, is computed from an exponential distribution with $\sum_s a_s$ as its parameter. The time is advanced by $\Delta\tau$ and the transition is effected. If $m = i$ for the transition, then n_i is decremented and n_j is incremented either immediately, as in the the transition from assessor to recruiter, or at a deterministic time in the future that represents the completion of the ant's navigation between sites. When $m \neq i$, then n_m, the number of ants in a recruiter state, is decremented to reflect the start of a tandem run or transport. If any recruitees are available at the time when the recruiter is expected to arrive at their site, then their population is decremented in the state counter. At the end of the recruiter's round-trip journey, the counter is updated to reflect the recruiter's success or failure at bringing another ant to the site. Whenever the counter is updated, the propensities must be recalculated and a new transition and $\Delta\tau$ are computed.

Microscopic Level. At the microscopic level, each ant is represented as an individual entity that stores knowledge of its task state, home nest, another nest site, position, speed, type of ant it is recruiting (if any), and whether it is navigating to a site. The stochastic simulation method described in the previous section is used to generate state transitions and their times. At this level, the simulation runs in time steps Δt to implement the ants' incremental navigation through their environment. As a result, the completion of inter-site navigation is checked at the beginning of every time step rather than acknowledged at the exact time it happens, and transitions at time τ are initiated when $t \leq \tau \leq t+\Delta t$.

When a transition is generated, a random ant in the appropriate task state that is not already en route to a site is selected to attempt recruitment or change state, either immediately or after traveling. Navigation functions [14] are used to generate ant trajectories that mimic the behavior of traveling between sites while avoiding obstacles. A navigation function provides a form for a feedback controller that guides an agent to a goal, the unique minimum of the function, while preventing collisions with obstacles. It can be defined on any environment that is deformable to one with a spherical boundary and disjoint, spherical obstacles.

In the simulation, ants and their destinations are represented as points, and obstacles are circular. The position r of an ant is updated at each time step by numerically integrating the equation

$$\dot{r} = -v\nabla\varphi_\kappa(r, r_d)/ \parallel \nabla\varphi_\kappa(r, r_d) \parallel \ , \tag{6}$$

where v is the ant's speed and $\varphi_\kappa(r, r_d)$ is the navigation function with the ant's current destination r_d. The φ of each ant share a common parameter κ, which was selected empirically. Various combinations of v and r_d are used to produce different agent controllers; for example, one $l \in L_a$ would be navigating from site 0 to site 1 at the tandem-running speed.

The microscopic simulation uses a centralized approach, since a "global planner" initiates transitions. However, the simulation has a decentralized equivalent: it produces transition times according to the same probability distribution as a strategy in which each ant, at every time step Δt, independently undergoes one of its possible transitions s with probability $c_s \Delta t$. To determine whether it can execute switch-dependent transitions, an ant only needs to know whether the population at its current site, P_q, exceeds a quorum. In a robotic system, this estimate can be achieved through local sensing. The advantage of the centralized simulation is its speed; unlike the decentralized approach, it does not require looping through all ants at each time step.

5.2 Application to the House-Hunting Model

We implemented macro-continuous, macro-discrete, and microscopic simulations in Matlab of the model (1)–(5). The model is reduced to the scenario of a destroyed home and two available new nests, although it is more detailed than the model in [1]. All ants are initially located at site 0, and all active ants begin as naive ants. The rate units are min^{-1}. The nests are 65 cm apart, the inter-site distance used in experiments to derive the site discovery and recruitment rates [16].

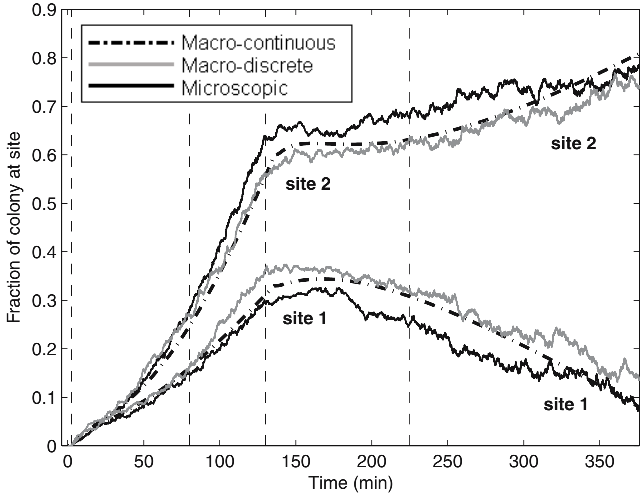


Fig. 4. Population fractions at sites 1 and 2; $p = 0.25$, $Q = (10/208)N$, $\mu_1 = \mu_2 = 0.013$, $\lambda_1 = \lambda_2 = 0.033$, $\rho_{12} = 0.008$, $k_1 = 0.016$, $k_2 = 0.020$, $\phi_1 = \phi_2 = 0.099$ (values are from [1], [16]); $\rho_{21} = 0.002$, $\kappa = 2.7$. Dashed vertical lines correspond to the times of the snapshots in Figure 5.

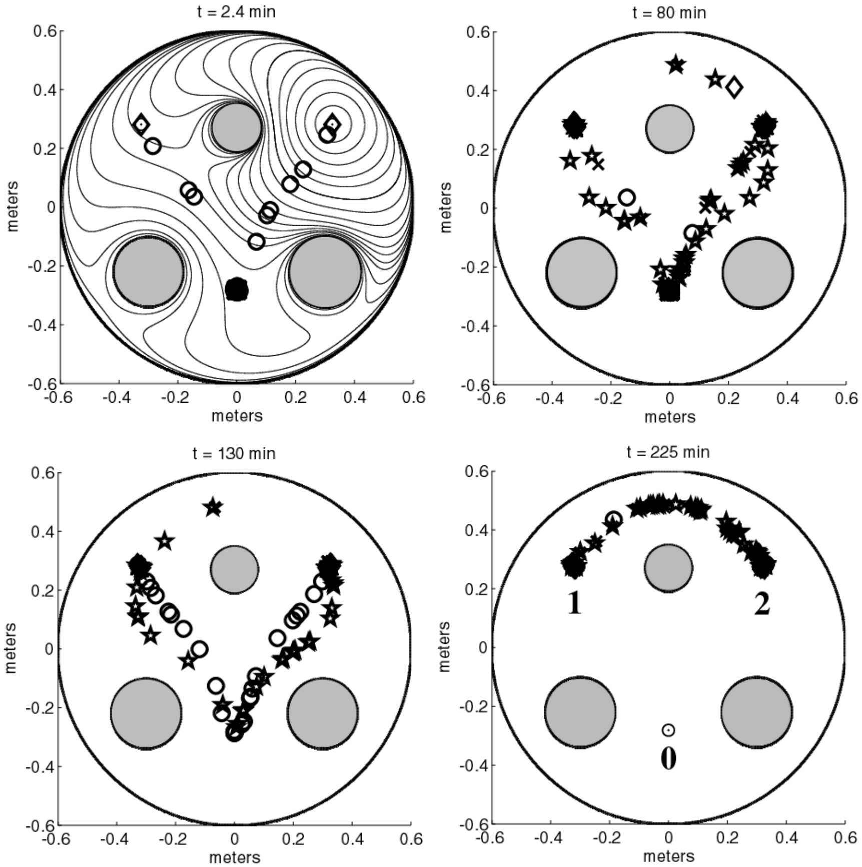


Fig. 5. Agent simulation snapshots (○ = *naive*; ◇ = *assessor*; ★ = *recruiter*; × = *passive*) showing the colony at (a) 2.4 min (top left); (b) 80 min (top right); (c) 130 min (bottom left); and (d) 225 min (bottom right). The navigation function that corresponds to an agent controller with r_d at site 2 is shown at the top left.

Each nest is represented as a circle of radius 0.02 m; an ant is considered inside the nest once it enters the circle. Ants performing tandem runs move at 1.5 mm/sec, while all other ants move at 4.6 mm/sec, the transport speed [1]. There are three obstacles in the environment. In the macro-continuous and macro-discrete simulations, the time delays due to navigation, measured from the microscopic simulation, are $\boldsymbol{\tau_{01}} = \boldsymbol{\tau_{02}} = 6$ min, $\tau_{\mathit{01}} = \tau_{\mathit{02}} = 2.2$ min, $\tau_{\mathit{10}} = \tau_{\mathit{20}} = 2.5$ min, $\boldsymbol{\tau_{12}} = \boldsymbol{\tau_{21}} = 7.84$ min, and $\tau_{\mathit{12}} = \tau_{\mathit{21}} = 2.48$ min.

Figure 4 displays the population fractions at sites 1 and 2 from the macro-continuous model solution and from macro-discrete and microscopic simulations with $N = 832$, $\Delta t = 0.05$ min. The two simulations match the macro-continuous model fairly well. Although not shown, it has been verified that the macro-discrete simulation approaches the macro-continuous model as N increases. In all plots, both sites achieve a quorum prior to 30 min and initially experience population growth. Site 2 outpaces site 1 in growth because ants commit to site 2 more quickly ($k_2 > k_1$) and are more willing to switch allegiance from site 1 to 2 than vice versa ($\rho_{12} > \rho_{21}$). By ∼130 min, all passive ants have been transported from site 0, and recruiters "forget" this site. The newly naive ants at site 1 or 2 repeat the process of finding, assessing, and recruiting to the other potential home site; however, now they can recruit *from* the site as well. Assessors at either site are more likely to recruit to the site of higher quality, which results in a net transport of passive ants to site 2. By ∼376 min in the macro-continuous and microscopic models, all passive ants at site 1 have been removed to be reunited with those at site 2; only active ants remain at site 1. Due to stochastic fluctuations, some passive ants still remain at site 1 in the macro-discrete model.

Figure 5 shows snapshots of the microscopic simulation at times indicated by the vertical lines in Figure 4. Nest sites are labeled in Figure 5d; gray circles denote obstacles. The curvature in the ant trajectories is due to the shape of the navigation functions, one of which is displayed in Figure 5a. The snapshots correspond to the initial searching and assessing phase (5a), the period of transport from site 0 (5b), the realization that site 0 contains no passive ants (5c), and the period of transport between sites 1 and 2 (5d).

6 Conclusion

We have described abstractions of a robotic swarm at three different levels and presented a methodology for synthesizing behaviors for individual robotic agents. Our behavioral synthesis at the highest level was derived from a mathematical model of an ant population. The macro-continuous model was reduced to a macro-discrete model to account for an integer number of agents. The macroscopic behaviors were then further realized by behaviors for individual agents. The components of the methodology have been illustrated through the analysis and simulation of nest site population growth in a model of ant house hunting.

We are interested in designing macroscopic swarm behaviors that are more relevant to multi-robot applications such as surveillance, sampling, and search-and-rescue. We want to determine parameter ranges and/or control terms that produce a desired group objective by using reachability analysis and control synthesis at the macro-continuous level. Here the methodology of [13] can be used to synthesize macro-continuous behaviors, which can be translated to macro-discrete level and then to microscopic level behaviors. We have begun this investigation in [18], where we adapt the house hunting model to a robot deployment task in which the swarm splits between two sites in a predefined ratio.

Acknowledgements. We are grateful for the support of NSF grants CCR02-05336 and IIS-0427313, and ARO Grants W911NF-05-1-0219 and W911NF-04-1-0148. We thank the workshop organizers and attendees for their helpful comments.

References

1. Franks, N., Pratt, S., Mallon, E., Britton, N., Sumpter, D.: Information flow, opinion polling and collective intelligence in house-hunting social insects. Phil Trans Roy Soc London B **357** (2002) 1567–1584
2. Berman, S., Halász, Á., Kumar, V.: MARCO: A reachability algorithm for multi-affine systems with applications to biological systems. Accepted to HSCC'07, Pisa, Italy, April 2007.
3. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York (1999)
4. Belta, C., Kumar, V.: Abstraction and control for groups of robots. IEEE Transactions on Robotics **20:5** (2004) 865–875
5. Martinoli, A., Easton, K., Agassounon, W.: Modeling of swarm robotic systems: a case study in collaborative distributed manipulation. Special issue on Experimental Robotics, Int. Journal of Robotics Research **23**(4) (2004). B. Siciliano (ed.) 415–436
6. Agassounon, W., Martinoli, A., Easton, K.: Macroscopic modeling of aggregation experiments using embodied agents in teams of constant and time-varying sizes. Autonomous Robots **17**(2-3) (2004). M. Dorigo, E. Sahin (eds.) 163–191
7. Lerman, K., Jones, C., Galstyan, A., Mataric, M.: Analysis of dynamic task allocation in multi-robot systems. Int. J. of Robotics Research, **25**(4) (2006) 225–242
8. Lerman, K., Martinoli, A., Galstyan, A.: A review of probabilistic macroscopic models for swarm robotic systems. In *Swarm Robotics Workshop: State-of-the-art Survey*, LNCS **3342** (2005). E. Sahin, W. Spears (eds.) 143–152
9. Gillespie, D.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J Comp Physics **22** (1976) 403–434
10. Crespi, V., Galstyan, A., Lerman, K.: Comparative analysis of top-down and bottom-up methodologies for multi-agent systems. Proc 4th Int'l Conf on Autonomous Agents and Multi Agent Systems (AAMAS'05), Utrecht, The Netherlands
11. Julius, A., Halász, Á., Kumar, V., Pappas, G.: Finite state abstraction of a stochastic model of the lactose regulation system of *Escherichia coli*. IEEE Conf on Decision and Control, San Diego, CA, December 2006. To appear.
12. Heylighen, F.: The science of self-organization and adaptivity. In The Encyclopedia of Life Support Systems, EOLSS Publishers Co. Ltd. (2003)
13. Habets, L., and van Schuppen, J.: A control problem for affine dynamical systems on a full-dimensional polytope. Automatica **40** (2004) 21–35
14. Rimon, E., Koditschek, D.: Exact robot navigation using artificial potential functions. IEEE Transactions on Robotics and Automation **8**(5) (1992) 501–518
15. Pratt, S., Sumpter, D., Mallon, E., Franks, N.: An agent-based model of collective nest choice by the ant *Temnothorax albipennis*. Animal Behav **70** (2005) 1023–1036
16. Pratt, S., Mallon, E., Sumpter, D., Franks, N.: Quorum sensing, recruitment, and collective decision-making during colony emigration by the ant Leptothorax albipennis. Behav Ecol Sociobiol **52** (2002) 117–127
17. Belta, C., Finin, P., Habets, L., Halász, A., Imielinski, M., Kumar, V., Rubin, H.: Understanding the bacterial stringent response using reachability analysis of hybrid systems. HSCC '04, Philadelphia. LNCS **2993**. R. Alur, G. J. Pappas (eds.) 111–125
18. Berman, S., Halász, A., Kumar, V., Pratt, S.: Bio-inspired group behaviors for the deployment of a swarm of robots to multiple destinations. Accepted to ICRA'07, Rome, Italy, April 2007.
19. Parrish, J., Hamner, W. (eds.): Animal Groups in Three Dimensions. Cambridge University Press, New York (1997)