Aggregation-mediated Collective Perception and Action in a Group of Miniature Robots

Grégory Mermoud Loïc Matthey William C. Evans Alcherio Martinoli
Distributed Intelligent Systems and Algorithms Laboratory
EPFL-ENAC-IIE-DISAL
École Polytechnique Fédérale de Lausanne
Lausanne, Switzerland
{gregory.mermoud, loic.matthey, william.evans, alcherio.martinoli}@epfl.ch

ABSTRACT

We introduce a novel case study in which a group of miniaturized robots screen an environment for undesirable agents, and destroy them. Because miniaturized robots are usually endowed with reactive controllers and minimalist sensing and actuation capabilities, they must collaborate in order to achieve their task efficiently. In this paper, we show how aggregation can mediate both collective perception and action while maintaining the scalability of the algorithm. First, we demonstrate the feasibility of our approach by implementing it on a real group of Alice mobile robots, which are only two centimeters in size. Then, we use a combination of both realistic simulations and macroscopic models in order to find optimal parameters that maximize the number of undesirable cells destroyed while minimizing the impact on the healthy population. Finally, we discuss the limitations of these models, both in terms of accuracy, computational cost, and scalability, and we outline the importance of an appropriate multi-level modeling methodology to ensure the relevance and the faithfulness of such models.

Categories and Subject Descriptors

I.2 [**Computing Methodologies**]: Artificial Intelligence; I.2.9 [**Artificial Intelligence**]: Robotics—*Autonomous vehicles*

General Terms

Algorithms, Performance, Experimentation, Theory

Keywords

Multi-robot systems, distributed problem solving, swarm intelligence, multi-level modeling

1. INTRODUCTION

Scientific and technological breakthroughs in the field of nano- and microengineering have steered the robotics community towards the realm of extreme miniaturization. Very small robots can access environments that are beyond the reach of typical robotic platforms, with recent case studies exploring scenarios such as the inspection of the human digestive tube [16] or complicated industrial machinery [2]. Further miniaturization down to the micro- or nanoscale holds even more exciting promises in a large variety of fields. However, miniaturization comes at a price: minimalist computational, sensing, actuation, and communication capabilities. These severe restrictions create the need for a collaborative approach towards the solving of tasks by leveraging both collective perception and action.

In this paper, we intend to show how self-organization and collective decision-making can be exploited to overcome the intrinsic limitations in terms of sensing and actuation of simplistic agents such as the miniaturized Alice mobile robot [5]. We introduce a case study in which a group of Alice robots must achieve collaborative screening of an environment in order to identify and destroy undesirable cells.

One can draw an analogy with different natural systems that are responsible for identifying and neutralizing pathogens in a given environment (e.g., the human immune system, or bacteria purifying environmentally polluted regions). Importantly, this task must be carried out in a reliable manner: the system must attack pathogens while preserving healthy actors of the environment. Similarly, in our case study, the environment contains two types of spots ("good" and "bad"), which differ from each other in an observable fashion. In Nature, pathogens are generally identified by reading from chemical receptors located at their surface; here, since cells are represented by colored spots displayed by an overhead projector, their distinctive property is light intensity, which can be perceived using the robots' onboard light sensor (see Figure 1(b)). However, light intensity measurements are clouded by both the intrinsic noise of the photocell, and the lighting variations of the projector, thus making the identification of the cells' type unreliable. In this paper, we show how collective perception and decision-making enable even simplistic robots to overcome their individual sensing limitations, and enhance the performance of the group, i.e., maximizing the destruction rate of "bad" spots while preserving "good" spots. In particular, we outline how a simple and reliable mechanism to share information among the robots is needed in order to achieve collective decision-making.

Aggregation allows us to solve this problem by replacing the transmission of a message by a physical contact (which can be thought also as a form of communication). In this paper, we use local IR beaconing merely as a way of discrimi-

Cite as: Aggregation-mediated Collective Perception and Action in a Group of Miniature Robots, Grégory Mermoud, Loïc Matthey, William Christopher Evans and Alcherio Martinoli, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. 599-606

(a) (b)

Figure 1: (a) Picture of an on-going real experiment with 5 robots and 4 spots. (b) Close-up of an Alice 2002 robot, which has a size of 2cm×2cm×2cm and is equipped with four infrared sensors for environment sensing and communication as well as an extension board with one photocell and two LEDs of different color (red and green) for tracking purpose.

nating between obstacles and other robots, a mechanism easily replicated at smaller length scales (e.g., by an electrical contact or pressure sensor). At the same time, aggregation is particularly scalable as it can mediated by a broad variety of interactions, such as capillary or electromagnetic forces, which are available down to the nanometer scale. In our case, the destruction of cells is triggered by tracking software that monitors the system, but in reality, the robots would need an actual cell-destruction mechanism, which can come in many forms. It is possible that in many scenarios more than one robot may be required to successfully destroy a cell, either because the effect of a single individual's action is too small or because the destruction mechanism is too complex (e.g., chemical reaction involving more than one reactant). Aggregation allows even minimalist robots to achieve these tasks in a self-organized manner, without relying on centralized guidance or complex coordination schemes. Note that this type of distributed control strategy requires efficient model-based analysis and synthesis methodologies to be successful; depending solely on intuition and trial-and-error schemes is not an option, especially from a scalability standpoint.

However, the complexity exhibited by stochastic aggregation and self-assembly prevents a single model from probing the dynamics of the whole system. These difficulties motivate a combination of multiple levels of abstractions, ranging from realistic physics-based simulations up to macroscopic models, into a consistent multi-level modeling framework. On the one hand, one needs microscopic models that are able to capture low-level details, namely robot direction, shape, and trajectory. On the other hand, one is interested in models that can yield accurate numerical predictions of collective metrics and investigate, possibly formally, macroscopic properties such as the size, the type and the proportion of the resulting aggregates. In this paper, we use a combination of both realistic simulations and macroscopic models in order to accomplish these goals. We discuss the limitations of macroscopic models, and we outline the importance of an appropriate multi-level modeling methodology to ensure the relevance and the faithfulness of such models. Finally, we discuss the scalability of these models with respect to the number of robots and spots; this feature is essential to the tractability of systems that involve large numbers of very small robots.

2. STATE-OF-THE-ART

Aggregation is an efficient mechanism exploited in nature that allows interactions and information exchange between individuals, enabling the emergence of complex collective behaviors [3] ranging from predator protection [17] to collective decision-making [5].

In robotics, aggregation has been extensively studied as a model of animal behaviors [3]. Mixed robot-animal societies have also proven very useful for studying and controlling self-organized behavioral patterns mediated by aggregation in group-living animals [5]. Aggregation also forms the basis for more complex tasks in distributed robotics, such as the assembly and the control of modular robots [16] or self-assembling robots [9, 14]. It has also served as as a benchmark for spatial models [6].

Aggregation is the underlying mechanism of a broad variety of distributed robotic systems where long-range communication is unfeasible or where a particular task requires more than one robot to collaborate in a coordinated manner. For instance, the stick-pulling experiment described in [12] relies on stochastic aggregation between pairs of robots, which are unable to achieve the task (i.e., pulling a stick out of a hole in the ground) alone. The present paper investigates a variation of this experiment in which the robots must not only achieve collaborative manipulation, but also collaborative perception and decision-making.

From a modeling perspective, both aggregation (e.g., [1]) and self-assembly (e.g., [9]) have been extensively studied using probabilistic models. Deterministic models of aggregation and flocking (which is conceptually similar to aggregation, but involves coordinated motion of the aggregate) such as [7] as well as graphical models of multi-robot systems such as [18] are interesting approaches from a system and control perspective, but do not take explicitly into account the intrinsic randomness of self-assembly processes. More recently, a series of works [9, 13, 14] proposed macro-continuous models of group robotic systems based on the Chemical Reaction Network (CRN) framework, and simulated using Gillespie's exact algorithm (or its optimized variations) [4]. The present paper discusses in further detail the importance of proper modeling for distributed robotic systems; in particular, it outlines the limitations of both deterministic ODE models and stochastic simulations.

3. MATERIALS AND METHODS

3.1 Case study and algorithm

The case study investigated in this paper involves collective decision-making in a group of minimalist robots, namely the Alice mobile robot, which is described in further details in Section 3.2. The environment is populated with N_s spots which can be either *good* or *bad* and N_r robots whose goal is to seek and destroy bad spots while preserving the good ones. Each time a spot is destroyed, it is immediately recreated at another location within the environment. In order to have a quantitative method of reporting system performance, we define a metric function M in terms of the number of good and bad spots destroyed:

$$M(\alpha) = \frac{D_{bad}}{(D_{good})^{\alpha} + 1} \quad (1)$$

where D_{bad} is the number of bad spots destroyed, and D_{good} is the number of good spots destroyed. The coefficient α

may be balanced according to the penalty one wishes to associate with the destruction of a good spot; the higher the coefficient, the higher the penalty. Hereafter, we always set $\alpha = 2$ since we want to emphasize the ability of the group to discriminate between good and bad spots.

In our current setup, the spots are colored circles of diameter d_{spot} drawn on an arena by an overhead projector; good spots are green and bad spots are red. The robots are equipped with a light sensor that can be used to assess a spot's type. However, the measure provided by the light sensor is noisy (see Figure 2(a)); therefore, the robots may mistakenly trigger the destruction of a good spot. We denote $p_{w,good}$ the probability that a robot believes a good spot to be a bad one (false positive) and $p_{w,bad}$ the probability that a robot believes a bad spot to be a good one (false negative). Depending on the distribution of light sensor measurements, these probabilities can be different. Since we assume the robots to be purely reactive, they form their belief on the basis of a single measurement and in a purely deterministic manner, by using a simple decision threshold whose value is $t_d = (\mu_{good} + \mu_{bad})/2$. We assume that the robots can always determine whether they are exploring a spot or not in a perfect manner.

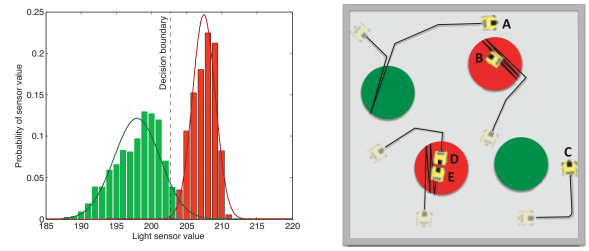
As mentioned earlier, collective decision-making is a way of overcoming the limitations of the individuals in terms of sensing. The question is, how can we achieve collective decision-making without communication? Here, we exploit aggregation as an implicit communication scheme that allows the robots, uniquely through their physical presence, to share their estimate of the type of the spot they are in. When two robots encounter each other in a spot, they form an aggregate only if both have the same estimate (e.g., this spot is a bad spot); otherwise they perform obstacle avoidance, and eventually leave the spot. Consequently, no aggregate is formed if the robots have differing estimates.

Therefore, one important parameter of our controller is k, which denotes the number of aggregated robots required to trigger the destruction of a spot. For $k = 1$, there is basically no collaboration: a single robot can destroy the spot it is exploring. For $k = 2$, the spot is destroyed as soon as a robot aggregates with another robot (Figure 2(b) depicts a typical experiment with $k = 2$). For $k = 3$, an aggregate can remain in a spot for a while without triggering its destruction, which therefore introduces a further parameter $p_{leave,aggr}$, that is, the probability that a robot leaves the aggregate it is part of. Note that, while we do not study models for $k > 3$, these are relatively easy to derive from the model for $k = 3$.

The optimal value of k depends on the difficulty of the task, i.e., the amount of noise characterizing the light sensor readings as well as their separability. Note that even in absence of noise, i.e., with $p_{w,good} = p_{w,bad} = 0$, more than one robot may be required to trigger the destruction of a spot (e.g., when individual robots are too small or limited for carrying out the task on their own).

As mentioned earlier, we assume our robots to be endowed with a reactive controller (Figure 3), and minimalist capabilities. Namely, nearby robots can detect each other (and distinguish other robots from obstacles), but they cannot actually communicate, nor can they carry out complex computation. Also, the robots do not have any reference to a global coordinate system. The robots explore the environment by performing a simple random walk with collision avoidance. After entering a spot, a robot will remain inside of it, "bouncing" off of its edges by performing a U-turn. At each bounce, a robot will decide to leave the spot with probability $p_{leave,good}$ or $p_{leave,bad}$, depending on whether it is exploring a good or bad spot, respectively. When a robot encounters an obstacle while exploring a spot, it assumes that it is another robot, and stops in order to form an aggregate. A robot may leave an aggregate with a probability $p_{leave,aggr}$, if it does no longer detect a nearby obstacle, or if the spot it is in was destroyed.

Since the aim of the group is to destroy bad spots while preserving good spots, typical values for these probabilities are $p_{leave,good} \approx 1$ and $p_{leave,bad} \approx 0$. While this intuition is optimal in some cases, it may prove suboptimal when collaboration is introduced (when $k > 1$). Table 1 summarizes the different parameters of the system as well as their default value.



(a) (b)

Figure 2: (a) Histogram of light sensor measurements (1000 values) in good spots (in green, left), and in bad spots (in red, right). Fitted gaussian distributions are also shown (continuous lines). The scenario of low difficulty is based on these data, with $p_{w,good} = 0.0743$ and $p_{w,bad} = 0.0017$. Hereafter, we also study a scenario of medium difficulty with $p_{w,good} = 0.271$ and $p_{w,bad} = 0.2224$. (b) Sketch of a typical experiment with 4 spots and 5 robots for $k = 2$. Trajectories of the robots are denoted by black lines. Robot A explored a good spot, made one wrong decision, but eventually left the spot. Robot B is exploring a bad spot, waiting for one mate. Robot C avoided an obstacle while exploring the environment. Robots D and E encountered each other in a bad spot, and decided to aggregate; this spot is therefore about to be destroyed, and re-located at some other location in the arena.

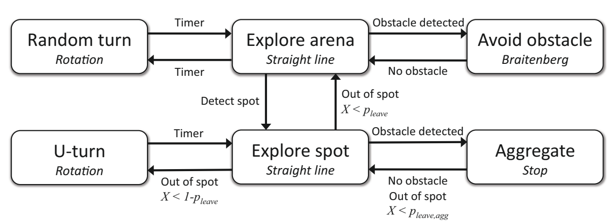


Figure 3: Schematic overview of the robots' controller. Each state has an associated action (in italic). Some transitions are probabilistic; in this case, the value of p_{leave} is either $p_{leave,good}$ or $p_{leave,bad}$ depending on the robot's estimate.

Table 1: Parameters of the system, their domain of definition, and their default value (if not stated otherwise).

Parameter	*Domain*	*Default value*
N_{robots}	$\mathbb{N}^*$	5
N_{spots}	$\mathbb{N}^*$	4
k	$\mathbb{N}^*$	Always specified
$p_{leave,good}$	$[0,1]$	1.0
$p_{leave,bad}$	$[0,1]$	0.0
$p_{leave,agg}$	$[0,1]$	0.001 (only for $k>2$)
$p_{w,good}$	$[0,1]$	0.271 (medium)
$p_{w,bad}$	$[0,1]$	0.2224 (medium)

3.2 Experimental Setup

The Alice 2002 mobile robot has a size of 2cm×2cm×2cm, a differential wheel drive that reaches speed up to 4cm/s, and four infrared sensors that allow the detection of passive obstacles at ranges up to 3cm simultaneously with 4bps bidirectional communication up to 6cm. The testing environment is a 50cm square arena. Local infrared communication allows aggregated robots to exchange some bits of information, albeit with poor reliability. In this case study, we assume the robots to be unable to implement an actual communication protocol; instead, we use local infrared communication merely as a way of discriminating between obstacles and other robots. Furthermore, robots are equipped with one epoxy-encapsulated photo-sensitive sensor (A 9950 11 photocell, Perkin Elmer, maximal spectral sensitivity at 530 nm) that allows robots to determine if they are inside of a spot and form an estimate of its type.

Obviously, robots cannot interact physically with spots that are pictured on the arena by an overhead projector. To overcome this problem, we employ one overhead camera in conjunction with SwisTrack, an open-source object tracking tool targeted to multi-agent systems [11]. In order to obtain an accurate measure of both the position and the orientation of the robots, we use markers that consist of two LEDs of different colors (red and green). Our system constantly integrates the trajectory of each robot and updates the environment accordingly, i.e., it detects aggregates, and modifies the display by relocating destroyed spots.

We carried out two types of experiments with $k = 1$ and $k = 2$; each experiment involved 5 robots and 4 spots, and was monitored automatically using the system described above. We repeated each experiment three times; all runs were performed in darkness, in order to minimize perturbations due to environmental lightning. The main objective of these experiments was to demonstrate the performance gain provided by collaboration.

3.3 Realistic simulation

We additionally implement the above experimental setup and hardware in Webots [15], a realistic simulator that is able to accurately model the non-linear sensor characteristics of the Alice robot, including noisy response of the sensors as well as wheel slip. Webots is particularly useful because it allows us to perform fast, automatic data collection and analysis over various parameter sets. For our systematic experiments, we used a computational cluster of 50 machines, each with an Intel Pentium 4 3.00 GHz and 1GB RAM. Transitions between experiment states (i.e., changes in the total number of aggregates of each size) were stored on disk for later analysis. Our Webots simulations execute over ten times faster than real time, and system analysis is much easier as we have access to the internal state of the robots. Hereafter, we employ Webots simulations as a reference for evaluating both qualitatively and quantitatively the faithfulness of our models at higher abstraction levels.

3.4 Macroscopic models

Herefater, we construct a series of macroscopic models where interactions between robots and spots are modeled as a Chemical Reaction Network (CRN). A set of reactions can be represented as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The set of vertices $\mathcal{V}$ represents the *complexes*, i.e., state variables that denote the number of robots in a given state. The set of directed edges $\mathcal{E}$ represents the *reactions* between complexes, i.e., the transition of a robot from one state to the other. Each reaction can be denoted by an ordered pair $(i, j) \in \mathcal{V} \times \mathcal{V}$, meaning that one complex i transitions complex j. Furthermore, each reaction is associated with a rate constant.

Since we are interested in the ability of aggregation to deal with the noise in decision-making, we explicitly incorporate beliefs representation in our model. More specifically, the state of a robot is not only defined by its internal state, i.e., whether it is wandering, exploring a spot, or part of an aggregate, but also whether its internal state is a correct representation of the reality. Note that, even though obstacle avoidance is a state of the robot, it is not explicitly modeled as such; rather, we assume that a robot seamlessly switch between wandering and obstacle avoidance states. Hereafter, we present three models, each corresponding to a different value of k. Since these models are built in an incremental fashion, we shall start by describing in details the model for $k = 1$, which is the basis for other models where $k > 1$.

3.4.1 Model for $k = 1$

In the case of $k = 1$, a robot can trigger the destruction of a spot on its own. Therefore, we shall distinguish between the following state variables:

- Robots searching for spots: X_o
- Robots in spot i with a correct estimate of the type of i: $X_{c,i} \quad \forall i \in \mathcal{S}$
- Robots in spot i with a wrong estimate of the type of i: $X_{w,i} \quad \forall i \in \mathcal{S}$
- Number of destructions of spot i: $X_{d,i} \quad \forall i \in \mathcal{S}$

where $\mathcal{S}$ denotes the set of spots in the system. Equation 2 provides a complete view of the $6 \cdot |\mathcal{S}|$ reactions of the CRN.

$$\begin{aligned} &X_o \underset{l_i^c + s_i^c}{\overset{e_i^c}{\rightleftharpoons}} X_{c,i}, \qquad X_o \underset{l_i^w + s_i^w}{\overset{e_i^w}{\rightleftharpoons}} X_{w,i}, \\ &X_{c,i} \xrightarrow{s_i^c} X_{d,i}, \qquad X_{w,i} \xrightarrow{s_i^w} X_{d,i}, \qquad \forall i \in \mathcal{S} \end{aligned} \tag{2}$$

where e_i^c and e_i^w are the rates at which a robot encounters a spot i and correctly or wrongly identifies its type, respectively. Similarly, l_i^c and l_i^w is the rate at which a robot leaves the spot it is exploring and believes to be of a given type, either correctly or wrongly, respectively, without destroying it. s_i^c and s_i^w are the same as l_i^c and l_i^w, except that the robot destroys the spot in this case.

3.4.2 Model for $k = 2$

Hereafter, we update the model for $k = 1$ in order to account for aggregation-mediated collaboration ($k = 2$). The fact that two robots are now needed to trigger the destruction of a spot has two main implications: (i) robots can change their belief about the type of the spot they are exploring at a rate m_i^w (from a correct to wrong belief), and m_i^c (from a wrong to a correct belief), and (ii) aggregation introduces non-linear reactions, i.e., reactions that involve more than one robot, whose rate is a_i. Equation 3 provides a complete view of the CRN.

$$\begin{array}{rlrl}
X_o \underset{l_i^c}{\overset{e_i^c}{\rightleftharpoons}} X_{c,i}, & & X_o \underset{l_i^w}{\overset{e_i^w}{\rightleftharpoons}} X_{w,i}, & \\
X_{c,i} \underset{m_i^c}{\overset{m_i^w}{\rightleftharpoons}} X_{w,i} & & 2\,X_{c,i} \xrightarrow{a_i} 2\,X_o, & \\
2\,X_{w,i} \xrightarrow{a_i} 2\,X_o, & & X_{c,i} + X_{w,i} \xrightarrow{2\,a_i} 2\,X_o & \\
2\,X_{c,i} \xrightarrow{a_i} X_{d,i}, & & 2\,X_{w,i} \xrightarrow{a_i} X_{d,i} & \\
X_{c,i} + X_{w,i} \xrightarrow{2\,a_i} X_{d,i} & & \forall i \in \mathcal{S} & \qquad (3)
\end{array}$$

3.4.3 Model for 3-Agg

If more than two robots are needed to trigger the destruction of a spot, it means that those robots that are part of a pair $X_{a,i}$ may remain idle in spot i, waiting for a third robot to join up, which is an event that happens at a rate t_i. The robots can also decide to leave the aggregate they are in at a rate l_i^a.

$$\begin{array}{rlrl}
X_o \underset{l_i^c}{\overset{e_i^c}{\rightleftharpoons}} X_{c,i}, & & X_o \underset{l_i^w}{\overset{e_i^w}{\rightleftharpoons}} X_{w,i}, & \\
X_{c,i} \underset{m_i^c}{\overset{m_i^w}{\rightleftharpoons}} X_{w,i}, & & 2\,X_{c,i} \xrightarrow{a_i} X_{a,i}, & \\
2\,X_{w,i} \xrightarrow{a_i} X_{a,i}, & & X_{c,i} + X_{w,i} \xrightarrow{2\,a_i} X_{a,i}, & \\
X_{c,i} + X_{a,i} \xrightarrow{2\,t_i} 3\,X_o, & & X_{w,i} + X_{a,i} \xrightarrow{2\,t_i} 3\,X_o, & \\
X_{a,i} \xrightarrow{l_i^a} 2\,X_o, & & X_{c,i} + X_{a,i} \xrightarrow{2\,t_i} X_{d,i}, & \\
X_{w,i} + X_{a,i} \xrightarrow{2\,t_i} X_{d,i}, & & \forall i \in \mathcal{S} & \qquad (4)
\end{array}$$

3.4.4 Simulation of Chemical Reaction Networks

We shall emphasize that CRNs merely provides a formal, yet detailed, description of the system at the macroscopic level; however, in order to yield quantitative predictions, CRNs need to be converted into either (i) macroscopic (macro-discrete) models, i.e., a continuous-time Markov processes whose states represent discrete numbers of robots, or (ii) macroscopic continuous (macro-continuous) models of ordinary different equations (ODE) whose state variables represent continuous fractions of the robotic population. However, these two approaches are not equivalent; while one can perform an exact simulation of macro-discrete models using Gillespie's algorithm [4], macro-continuous models of ODEs rely on an approximation, which we call hereafter the ODE approximation, that becomes exact if we scale the system such that the reaction rates become large and the effects of those reactions small, i.e., the system implies a large number of small changes. We discuss in further detail the validity of the ODE approximation in Section 4.2.

In this paper, we use the StochKit toolbox [10] in order to efficiently perform stochastic simulations of macro-discrete models. We convert CRNs into macro-continuous models using the relation $\dot{\mathbf{y}} = \mathbf{S} \cdot \mathbf{p}(\mathbf{y})$ where $\mathbf{S} = (s_{ij})$ is the stoichiometry matrix, with the stoichiometric coefficient s_{ij} of the j-th species in the i-th reaction, and is the propensity vector $\mathbf{p}(\mathbf{y})$, which depends on the reaction rates and the population of reactants for each reaction. Then, we numerically integrate this system of ODEs using MATLAB's `ode15s` function.

3.5 Rates identification

One of the crucial tasks when constructing macroscopic models is the identification of the different reaction rate constants. In our case, some of them are determined by using geometric approximations; other are measured using systematic Webots simulations. In order to use geometric approximations, we assume that our system is well-mixed, i.e., the probability that a robot is at a given position is independent of time and uniformly distributed over the arena space. Therefore, the encountering probability of a robot and another object (which, in our case, would be either another robot or a spot) is given by

$$p_c \sim \frac{\widehat{v}\, w_d}{A_{tot}} \qquad (5)$$

where $\widehat{v}$ is the average velocity of a robot, w_d is the diameter of the object, and A_{tot} is the total area of the region in which the robot is moving.

As a result, we can write

$$\left.\begin{array}{l} e_i^c = p_{c,spot} \cdot (1 - p_{w,good}) \\ e_i^w = p_{c,spot} \cdot p_{w,good} \end{array}\right\} \quad \text{if } i \text{ is a good spot}$$

$$\left.\begin{array}{l} e_i^c = p_{c,spot} \cdot (1 - p_{w,bad}) \\ e_i^w = p_{c,spot} \cdot p_{w,bad} \end{array}\right\} \quad \text{if } i \text{ is a bad spot}$$

with

$$p_{c,spot} = \frac{\widehat{v} \cdot d_{spot}}{A_{tot}} \qquad (6)$$

where $\widehat{v}$ is the average velocity of a robot, d_{spot} is the diameter of a spot, and A_{tot} is the area of the arena.

Similarly, the encountering rates between robots can be written

$$a_i = \sigma_a \frac{\widehat{v}\, d_{robot}}{A_{spot}} \qquad t_i = \sigma_t \frac{\widehat{v}\, 2\, d_{robot}}{A_{spot}} \qquad (7)$$

where $\widehat{v}$ is the average velocity of a robot, d_{robot} is the diameter of a robot, A_{spot} is the area of the spot i, and σ_i and σ_t are two parameters that account for the poor IR coverage of the Alice robot, which may sometimes prevent aggregation. We set hereafter $\sigma_i = 0.6$ and $\sigma_t = 0.5$; these values were measured in systematic Webots simulations.

The identification of the rates m_i^w, m_i^c, l_i^w, l_i^c, s_i^w, and s_i^c is slightly more difficult because they depend on the probability p_b that a robot encounters the border of the spot while exploring it. Using again a geometric approximation, we can write

$$p_b = \frac{\widehat{v}}{d_{spot}/2} = \frac{1}{T_t} \qquad (8)$$

where T_t is the average time taken by a robot to traverse a spot of diameter d_{spot}. Now, we can write

$$m_i^w = p_b \cdot (1 - p_{leave,good}) \cdot p_{w,good}$$
$$m_i^c = p_b \cdot (1 - p_{leave,bad}) \cdot (1 - p_{w,good})$$

$$l_i^w = p_b \cdot p_{leave,bad} \qquad s_i^w = p_b \cdot (1 - p_{leave,bad})$$
$$l_i^c = p_b \cdot p_{leave,good} \qquad s_i^c = p_b \cdot (1 - p_{leave,good})$$

where i denotes a good spot.

If i denotes a bad spot, $p_{leave,good}$, $p_{leave,bad}$, and $p_{w,good}$ must be changed to $p_{leave,bad}$, $p_{leave,good}$, and $p_{w,bad}$, respectively.

4. RESULTS AND DISCUSSION

4.1 Experiments with real robots

We present hereafter a series of experiments with real Alice robots in order to demonstrate the feasibility of our approach. Given the intrinsically stochastic nature of the investigated processes, a large number of runs is required in order to obtain statistically relevant data, which results in extremely time-consuming experiments if real robots are used. We provide experimental results (see Table 2) that suggest the relevance of collective perception and action as a mechanism for coping with unreliable sensing at the individual level. In spite of the high variability of the results obtained with real robots ($> 100\%$ of variability on the performance metric), collaboration seems to provide a non-negligible performance gain (as defined in Equation 1), up to two orders of magnitude in these particular experiments.

Furthermore, experiments with real robots provide crucial insights into the dynamics of the system, and features that one should account for in models at higher abstraction level. In particular, we observe that real robots tend to “loose” track of the spot they are exploring; because of noisy locomotion (due to both unreliable motors and wheel slip), robots do not perform exact U-turns, which can lead them to exit a spot involuntarily. Similarly, real robots tend to spend slightly more time around obstacles and walls than in the rest of the arena due to obstacle avoidance. While these effects are captured in realistic simulations, which account for spatiality, friction, and sensor and actuator noise, macroscopic models assume perfect mixing and uniform distribution of the robots throughout the environment.

4.2 Validation of macroscopic models

In order to validate our macroscopic models, we use realistic simulations (Webots) as a baseline. Figures 4(a) and 4(b) compare the predictions of both stochastic simulations (500 runs) and deterministic ODE model with those of Webots simulations (50 runs) with $N_r = 5$ and $N_s = 4$. Global, qualitative trends of all metrics are correctly captured by both models. Note that quantitative discrepancies in the performance metric are mainly due to the squared term of Equation 1. Also, macroscopic models do not account for a few effects observed in experiments with real robots, and captured by realistic simulations (see Section 4.1).

Experimental time, on the other hand, is greatly reduced as the level of abstraction increases. Stochastic simulations are four orders of magnitude faster than realistic simulations, which are in turn one order of magnitude faster than real time; ODE are roughly as fast as stochastic simulations (for a single run), but the former do not require multiple runs since they are solved using deterministic methods. These statistics are valid for small systems such as those investigated here; we discuss in further details the scalability of macroscopic models in Section 4.3.

The limits of the ODE approximation.

One observation from Figure 4(b) is that the ODE model is somewhat inaccurate for $k > 1$, due to (i) the small number of robots and spots involved in the system, and (ii) the compartmentation caused by spots, which act like vesicles in cell biology, i.e., they form weakly dependent subsystems characterized by even smaller numbers of robots, and whose reactions are usually very slow, and effect very large. The latter cause is arguably the most important one, and is also found in biology as a strong limitation of the use of macro-continuous models. In particular, the strong irregularity in the landscape of spot destructions around $p_{leave,good} = p_{leave,bad} = 0$ for $k = 3$ is not captured at all by the ODE model (Figure 4(b)), because it does not account for the situation where all robots are either aggregated or exploring different spots, thus leading to deadlocks. This result is similar to that obtained in the stick-pulling experiment [12], which involved, however, a deterministic timeout for controlling the waiting time rather than a leaving probability.

This type of situation is even better illustrated in scenarios where the robots have a perfect estimate of the type of the spots and the spots outnumber the robots (Figure 5). In this case, since there is no “intrinsic” randomness due to the noise, a robot with $p_{leave,bad} = 0$ may explore indefinitely a spot, waiting for a mate. If there are more spots than robots, the situation where each robot is waiting for another one in a different spot may arise, thus leading to a deadlock. As a result, the optimum shifts towards non-zero leaving probabilities as the ratio of spots to robots increases. This situation is not captured at all by the ODE model, which does not predicts any significant variation of the optimum.

In practice, however, the robots may involuntarily leave the spot they are exploring, essentially preventing leaving probabilities from being below a certain threshold, regardless of the actual value of $p_{leave,good}$ and $p_{leave,bad}$. As a result, the drop in performance predicted by stochastic simulations for very small leaving probabilities in the case high ratios of spots to robots is not observed in Webots simulations. However, the ODE model does not account for any drop in performance whatsoever; even worse, it predicts a slight increase of the performance as the number of spots increase. This finding emphasizes the importance of multi-level modeling when studying distributed systems; abstractions that seem correct *a priori* may actually be revealed to be error-prone. In our particular case, neglecting the effect of robots that “lose” track of the spot they are exploring leads to inaccuracies at the macroscopic level when leaving probabilities become very small. More importantly, the small number of robots and spots in the system as well as the compartmentation effect caused by spots render the ODE approximation invalid, and predictions of macro-continuous models very inaccurate, in particular when the ratio of spots to robots increases.

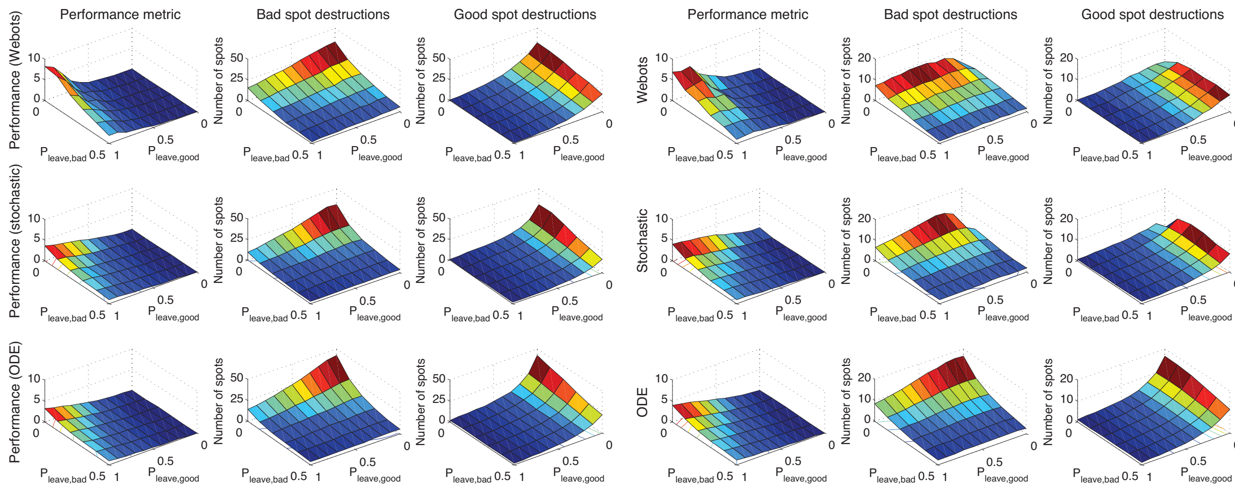
4.3 Scalability

Hereafter, we discuss the scalability of our models, i.e., their ability to capture the dynamics of systems that involve growing numbers of robots and spots while preserving affordable requirements in terms of memory and computation. We believe that, in the future, this problem will be crucial to the design and the control of massively distributed systems, composed of a multitude of ultra-small robots.

Table 2: **Summarized results of two experiments (with and without collaboration) using 5 real Alice robots and 4 spots (2 of each kind). Destruction rates are given in number of spots destroyed per minute. The performance of the swarm (Equation 1) is two orders of magnitude higher when collaboration is introduced.**

Destruction rate	Without collaboration ($k = 1$)		With collaboration ($k = 2$)	
	Bad spots	*Good spots*	*Bad spots*	*Good spots*
Run 1	4.93	3.85	0.68	0.09
Run 2	5.28	2.68	0.55	0.00
Run 3	5.12	2.95	1.56	0.20

Performance	Without collaboration ($k = 1$)	With collaboration ($k = 2$)
Run 1	$2.9 \cdot 10^{-2}$	2.8
Run 2	$7.0 \cdot 10^{-2}$	9.0
Run 3	$5.8 \cdot 10^{-2}$	1.76



(a) (b)

Figure 4: Predictions of Webots simulations (top), stochastic simulations (middle), and ODE models (bottom) for (a) $k = 2$, and (b) $k = 3$. For $k = 2$, the qualitative trends of the performance metric are correctly captured. The number of spot destructions are correctly predicted by both macroscopic models, even though ODE models tend to overestimate it for $p_{leave,good} = 0$. For $k = 3$, while both models exhibit excellent qualitative agreement, stochastic simulations yield much better quantitative predictions than ODE models, which not only overestimate the number of spot destructions, but are also unable to capture some of the irregularities that appear in the landscape of spot destructions, especially around $p_{leave,good} = p_{leave,bad} = 0$.

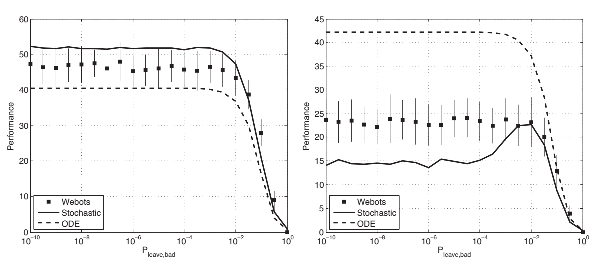


Figure 5: Predicted performance of the group with $k = 2$ for $N_{spots} = 4$ (left), and $N_{spots} = 10$ (right), and a constant number of robots $N_{robots} = 5$ that have a perfect estimate of the type of the spots. Stochastic simulations and the ODE model are consistent with Webots simulations when there are less spots than robots, but their predictions diverge for $N_{spots} = 10$.

Realistic simulations become rapidly intractable as the number of robots increases (as a rule of thumb, simulations with more than 50 robots become prohibitively expensive in terms of computational resources). At a slightly higher abstraction level, microscopic simulations (e.g., agent-based models) may represent a potential solution for intermediate group sizes, but they remain heavily memory and computation intensive.

Therefore, macroscopic models seem the most suitable tools to study massively distributed robotic systems. In our case, the number of reactions in the CRN is proportional to the number of spots N_s in the system, which therefore slows down both stochastic simulations and ODE models. Our experiments showed that simulation times scale in $O(N_s^3)$ for ODE models, and in $O(e^{N_s})$ for stochastic simulations. In terms of memory usage, ODE models also scale up much better than stochastic simulations. Stochastic simulations used 8GB of RAM for $N_s = 2000$ and $N_r = 4000$ whereas our ODE models only required 1GB for the same parameters. Finally, the accuracy of our ODE models is expected to increase as the number of robots N_r, and therefore the number of events, increases; indeed, we measured a 4-fold decrease of the mean square error of ODE models with respect to stochastic simulations by increasing N_s from 10 to 200 and N_r from 20 to 400. These findings motivate the use of ODE models for the study of very large distributed systems, or the use of hybrid approaches that preserve the stochastic behavior of the CRN, yet enable scalability [8].

5. CONCLUSION

In this paper, we presented a novel case study concerned with the screening of an environment by a group of miniaturized robots, whose goal is to seek and destroy bad spots while leaving good spots untouched. We proposed an algorithm that relies on collective perception and action in order to achieve this task with purely reactive robots. We demonstrated the feasibility of this approach by implementing it on a real group of Alice mobile robots. Building up models at different abstraction levels, starting with realistic simulations up to macroscopic models based on the Chemical Reaction Network (CRN) framework, we performed systematic searches of the parameter space that demonstrated the existence of optimal control parameters. An important finding of our research is that even simplistic robots endowed with very poor sensing capabilities can achieve a good performance in this type of case study by relying on collaboration.

Another very important conclusion of our work is the relevance of multi-level modeling for optimizing both the design and the control of distributed robotic systems, especially in the case of self-organized strategies such as the one employed in this paper. Indeed, on the one hand, realistic simulations are very useful tools in robotics, but they are prohibitively expensive for performing fine-grained systematic searches, or for investigating the dynamics of systems that involve a large number of robots. On the other hand, macroscopic models can be extremely fast, but their underlying abstractions (e.g., spatiality, ODE approximation) can sometimes lead to intolerable inaccuracies, if not validated properly. Multi-level modeling allows fulfillment of both requirements in a very efficient way by building up models at further abstraction levels in order to capture the relevant features of the system.

6. REFERENCES

[1] W. Agassounon, A. Martinoli, and K. Easton. Macroscopic modeling of aggregation experiments using embodied agents in teams of constant and time-varying sizes. *Autonomous Robots*, 17(2-3):163–192, Jan 2004.

[2] N. Correll and A. Martinoli. Multirobot inspection of industrial machinery. *IEEE Robotics & Automation Mag.*, 16(1):103 – 112, Mar 2009.

[3] S. Garnier, C. Jost, J. Gautrais, M. Asadpour, G. Caprari, R. Jeanson, A. Grimal, and G. Theraulaz. The embodiment of cockroach aggregation behavior in a group of micro-robots. *Artificial Life*, 14(4):387–408, Jan 2008.

[4] D. T. Gillespie. Stochastic simulation of chemical kinetics. *Annual Review of Physical Chemistry*, 58:35–55, Jan 2007.

[5] J. Halloy, G. Sempo, G. Caprari, C. Rivault, M. Asadpour, F. Tache, I. Said, V. Durier, S. Canonge, J. M. Ame, C. Detrain, N. Correll, A. Martinoli, F. Mondada, R. Siegwart, and J.-L. Deneubourg. Social integration of robots into groups of cockroaches to control self-organized choices. *Science*, 318:1155–1158, Jan 2007.

[6] H. Hamann, H. Worn, K. Crailsheim, and T. Schmick. Spatial macroscopic models of a bio-inspired robotic swarm algorithm. In *Proc. of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008)*, pages 1415–1420, 2008.

[7] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. on Automatic Control*, 48(6):988–1001, Jan 2003.

[8] T. Kiehl, R. Mattheyses, and M. Simmons. Hybrid simulation of cellular behavior. *Bioinformatics*, 20(3):316–322, Jan 2004.

[9] E. Klavins. Programmable self-assembly. *IEEE Control Systems Magazine*, 27:43–56, Jan 2007.

[10] H. Li, Y. Cao, L. R. Petzold, and D. T. Gillespie. Algorithms and software for stochastic simulation of biochemical reacting systems. *Biotechnol Progr*, 24(1):56–61, Jan 2008.

[11] T. Lochmatter, P. Roduit, C. Cianci, N. Correll, J. Jacot, and A. Martinoli. Swistrack - a flexible open source tracking software for multi-agent systems. In *Proc. of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008)*, pages 4004–4010, 2008.

[12] A. Martinoli, K. Easton, and W. Agassounon. Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *Int. J. Robotics Research*, 23(4-5):415–436, Jan 2004.

[13] L. Matthey, S. Berman, and V. Kumar. Stochastic strategies for a swarm robotic assembly system. In *Proc. of the 2009 IEEE Int. Conf. on Robotics and Automation (ICRA 2009)*, pages 1953–1958, May 2009.

[14] G. Mermoud, J. Brugger, and A. Martinoli. Towards multi-level modeling of self-assembling intelligent micro-systems. *Proc. of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, 1:89–96, May 2009.

[15] O. Michel. Webots: Professional mobile robot simulation. *Journal of Advanced Robotics Systems*, 1(1):39–42, 2004.

[16] Z. Nagy, R. Oung, J. Abbott, and B. Nelson. Experimental investigation of magnetic self-assembly for swallowable modular robots. In *Proc. of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008)*, pages 1915–1920, 2008.

[17] J. Parrish and L. Edelstein-Keshet. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science*, 284(5411):99–101, Jan 1999.

[18] G. A. S. Pereira, V. Kumar, and M. F. M. Campos. Closed loop motion planning of cooperating mobile robots using graph connectivity. *Robotics and Autonomous Systems*, 56(4):373–384, Jan 2008.