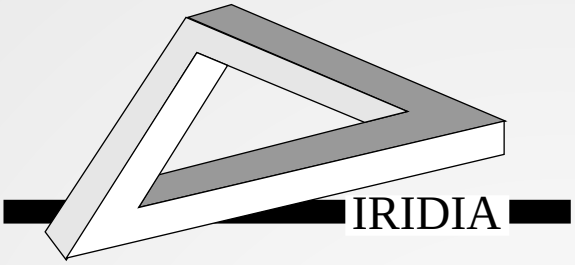


Modeling Pattern Formation in a Swarm of Self-Assembling Robots

V. TRIANNI, Th. H. LABELLA, R. GROSS,
E. SAHIN, M. DORIGO and J.-L. DENEUBOURG

Technical Report No.

TR/IRIDIA/2002-12
May, 2002

Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle

50 av. F. Roosevelt • CP 194/6 • B-1050 Bruxelles • Belgium
Phone: +32-2-650.27.29 • Fax: +32-2-650.27.15

Modeling Pattern Formation in a Swarm of Self-Assembling Robots

Vito Trianni[1], Thomas H. Labella[1], Roderich Gross[1], Erol Şahin[1], Marco Dorigo[1], and Jean-Louis Deneubourg[2]

TR/IRIDIA/2002–12
[1]IRIDIA, Université Libre de Bruxelles, Belgium
[2]CENOLI, Université Libre de Bruxelles, Belgium
{vtrianni,hlabella,rgross,esahin,mdorigo,jldeneub}@ulb.ac.be

Abstract. Self-assembly, the self-organized creation of structures composed of independent entities, represents a challenging class of problems for swarm intelligence. In this report, we present some preliminary results on pattern formation carried out by a group of simulated robots. We have defined a control architecture based on probabilistic choices of different basic behaviors. This architecture allows us to define the behavior of the robots, such that their interactions lead the group to form a desired pattern. In order to better understand the dynamics of the pattern formation task, we have applied a methodology composed of the following steps: first, the self-organizing process that leads to the formation of a particular pattern is investigated, producing a suitable mathematical model. Then, systematic experiments are conducted in simulation to estimate the parameters of the model. Finally, the predictions of the mathematical model are compared with the simulation results. This methodology is applied to the chain and cluster formation tasks, and the obtained results are reported.

1 Introduction

Self-assembly, the self-organized creation of structures composed of independent entities which are autonomous in their control, occurs in a wide range of natural systems ranging from chemistry to biology (for a review see [3]). A particularly interesting form of self-assembly is observed in social insects, which create different types of structures by physically attaching to each other [2, 1]: curtains, festoons, ovens, thermoregulatory clusters, swarms, bivouacs, bridges, doorways, pulling chains and rafts are all examples of self-assembling in social insects. Self-assembly represents a challenging class of problems for *swarm intelligence* [2]: the study of how collectively intelligent systems can be created by a number of simple autonomous agents.

Swarm intelligence is the main inspiration for the *Swarm-bots project*, within which the work presented in this report is carried out. The project aims to create a *swarm-bot*, a self-assembling and self-organizing artifact, composed of smaller mobile robots, called *s-bots*. The s-bots can aggregate to form a swarm-bot when necessary or the swarm-bot can split up into its s-bots when not.

Besides aggregation, a swarm-bot can change its shape on-line and has adaptive capabilities to adapt to its environment (see [4] for a more detailed description).

The swarm-bot concept lies between two main streams of robotics research: collective robotics [6, 5] and metamorphic robotics [9, 11]. In collective robotics, autonomous mobile robots interact with each other to accomplish a particular task. However, unlike s-bots, they do not have the ability to attach to each other by making physical connections. In metamorphic robotics, the system consists of connected self-contained modules that, although autonomous in their movements, remain attached to each other, lacking the full mobility of s-bots.

One of the main goals of the Swarm-Bots Project is the design of swarm intelligent control algorithms that let a group of autonomous mobile robots self-assemble. In this report, we focus on one of the basic tasks of self-assembling systems: spatial pattern formation. In particular, we study chain and cluster formation by a group of s-bots. These patterns are static in the sense that they do not move in the environment. Nevertheless, patterns are dynamic in the sense that each robot composing them is able to move, connect and disconnect thus modifying the pattern.

In order to understand the dynamics that leads the swarm of s-bots to self-organize in certain patterns, we propose a methodology based on the analytical modeling of global features of the pattern formation task [7]. This methodology consists of three main steps: first, a suitable analytical model describing the pattern formation task is defined. Second, the model parameters are estimated from systematic experiments of the simulated system. Finally, the global behavior predicted by the model is compared with the system. Once validated, the analytical model can be used to predict the group behavior under certain conditions, or it can serve as a feedback tool in the s-bot control design.

This work has been carried out in a simulated environment, described in Sec. 2. We have designed a control architecture that allows us to define the behavior of a single s-bot. The self-organized global pattern emerges from the numerous interactions between s-bots. The control model is based on probabilistic activation of basic behaviors, and it will be described in Sec. 3. Section 4 introduces the pattern formation task and describes the peculiarities of the chains and clusters formation. In this section, the behavior of a single s-bot is also detailed. Then, chain and cluster patterns are analyzed following the described methodology respectively in Sec. 5 and Sec. 6, and the obtained results are presented. Finally, the proposed methodology is discussed under the light of the experimental results in Sec. 7, while Sec. 8 concludes suggesting some future research directions.

2 Simulation

We have developed a simulator to study the swarm-bot control algorithms. Figure 1 shows three groups of s-bots[1] in the simulation environment. Each circle

[1] The mechatronic concept is described in more details in [8].

represents an s-bot. Two different types of s-bots are shown on the left and right side of the figure. The color of the ring shows the type of the s-bot. The rectangular tentacle placed on the body represents the gripper. The inner circle shown on the body denotes a controllable light placed on top of the s-bot for signaling.

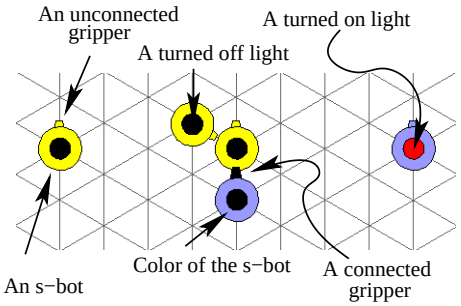


Fig. 1. Left and right side: two different types of disconnected s-bots. At the center, the lower s-bot gripped the upper one. This is indicated by a gripper that is longer and darker. The s-bot to the left remains unconnected to the two, although it is within gripping distance to the upper s-bot. A black inner circle represents a turned off top light (such as the one on the leftmost s-bot), and a lighter colored one represents a light that is on. The light of the rightmost s-bot in the figure is on, whereas the lights of the other s-bots are off.

The movement of the s-bot is modeled similar to that of a differential drive mobile robot. The control of the movement is discretized to guarantee that the s-bot is always positioned on the nodes of a hexagonal grid. This simplification made the implementation of connecting and disconnecting of the s-bots easier.

When not gripped, the s-bots can move freely in the environment guided by their sensors. However, when gripped, an s-bot loses its mobility. For instance, the upper s-bot, shown in a group of three at the center of Fig.1, is immobilized since it is gripped by the lower s-bot. All the other s-bots in the figure are free to move since they are not gripped by other s-bots. Physical pushes are not considered at this level of simulation.

The s-bots are equipped with different short-range and long-range sensing and signaling modalities. All the signaling and sensing is modeled as a light source/light sensor pair. The signal emitted by a source decreases in intensity with the square of the distance. The sources are characterized by the intensity, color, and the beam width of the signal. The top light[2] of the s-bot, described above, is an omnidirectional source. An s-bot type is signaled by its color. To make the orientation of an s-bot visible, three colored light sources are placed

[2] In the rest of the report, this is referred as the *light* that is turned on or off by the s-bot.

uniformly around the s-bot. These are called the left, right and the rear lights of the s-bot.

The sensors are directional and can sense the color and the intensity of the signal within their viewing angle. On the s-bots, short-range sensing is achieved by six proximity sensors, uniformly distributed around the body of the s-bot. They allow the s-bots to sense the presence of other objects and robots in the immediate vicinity. The long-range sensing is achieved by six directional light sensors, placed uniformly around the s-bot. Through all these sensors an s-bot can sense the top lights when they are on, the type as well as the orientation of other s-bots.

It is important to note that there is no explicit communication between the s-bots. The coordination between the s-bots takes place solely through their embodiment, the connections among them, and the signals emitted.

3 Control Model

The control architecture of an s-bot is based on the probabilistic activation of basic behaviors, such as the attraction or repulsion from light sources. It has been defined for the purpose of generalizing the definition of the s-bot behavior, being as much independent as possible from the given task. The s-bot control architecture is defined as a tuple:

$$\langle S, A, B, \mathcal{P}, s, h, p \rangle, \tag{1}$$

where

- S is the set of sensors,
- A is the set of actuator commands,
- B is the set of basic reactive behaviors $b : \mathbb{R}^{|S|} \to A$,
- $\mathcal{P}$ is a set of parameters, i.e. activation probabilities,
- $s : S \to \mathbb{R}$ is a *sensing* function that associates to each sensor its reading,
- $h : \mathbb{R}^{|S|} \to \mathbb{N}$ is a *context* function defining the current context the s-bot has perceived through its sensors as a natural number function of the given task.
- $p : \mathbb{N} \times \mathcal{P} \to B^3$ is a probabilistic activation rule that selects three basic behaviors at each time step.

Sensors. The set S contains all the sensors described in Sec. 2. In particular, each element is defined as a pair $\langle type, offset \rangle$, which refers respectively to the sensor type and the offset angle with respect to the s-bot heading, making it possible to differentiate between all the sensors.

Actuator Commands. As described in Sec. 2, an s-bot has essentially three actuators: movement, light and gripper. The set A refers to actuator commands that can be issued to actuators in order to perform a given action. It can be partitioned into three subsets, A_m, A_l and A_g, referring respectively to the movement, light and gripper commands.

Basic Behaviors. The set B accounts for basic behaviors. Even in this case, it can be partitioned into three subsets, B_m, B_l and B_g, referring respectively to movement, top light and gripper actuation. These subsets contain basic behaviors that are mutually exclusive, i.e. only one from each subset can be executed during the same time step. Behaviors belonging to different subsets can be executed in parallel, since they control different s-bot actuators. The movement behaviors (B_m) are:

- *Light Attraction* (LA): the s-bot moves in the direction of the light intensity gradient.
- *Light Repulsion* (LR): the s-bot moves in the opposite direction of the light intensity gradient.
- *Robot Attraction* (RA): the s-bot is attracted by the presence of other s-bots, detected by the proximity sensors distributed around the s-bot body. The proximity sensors have a limited sensing distance, thus the robot attraction vector relies only on local information.
- *Robot Repulsion* (RR): the s-bot is repelled by the presence of other s-bots.
- *Random Movement* (RM): the s-bot moves in a random direction.

The light behaviors (B_l) are:

- *Light On* (ON): the s-bot turns on the top light.
- *Light Off* (OF): the s-bot turns off the top light.

The gripper behaviors (B_g) are:

- *Gripper Open* (GO): the s-bot opens the gripper.
- *Gripper Close* (GC): the s-bot tries to connect a neighboring s-bot by closing its gripper.

All the subsets have a standard null behavior that keeps the state of the s-bot unchanged.

Parameters. The set $\mathcal{P}$ refers to control parameters. In particular, $\mathcal{P}$ refers to activation probabilities. Each basic behavior $b \in B$ has an associated activation probability $P_b(i) \in \mathcal{P}$, where $i \in \mathbb{N}$ is an index referring to the s-bot context, as it is defined by the h context function (see below). Thus, the activation probabilities depend on the context of the s-bot. The actual correspondence between the s-bot context and the activation probabilities is given by a matrix $\mathbf{P}$, where each row refers to a different context and each column to a particular basic behavior:

$$\mathbf{P} = \{P_{i,\hat{b}} = P_b(i),\ i = 1, \ldots, n,\ \hat{b} = 1, \ldots, |B|\}, \tag{2}$$

where $\hat{b}$ is an index referring to the basic behavior $b \in B$ and n is the number of possible contexts for a given task. The following properties must be satisfied:

$$\forall i,\ \sum_{b \in B_m} P_b(i) = 1, \qquad \sum_{b \in B_l} P_b(i) = 1, \qquad \sum_{b \in B_g} P_b(i) = 1, \tag{3}$$

Equation (3) accounts for the mutual exclusion of basic behaviors belonging to the same subset, assuring that one action is chosen at each time step from all the subsets.

Sensing Function. The sensing function s simply returns the sensor reading of each sensor $\langle type, offset \rangle \in S$, which is in general a real number. The sensing function applied on all the available sensors returns a reading vector $s(S) = \mathbf{S} \in \mathbb{R}^{|S|}$. This is used to determine the s-bot context $h(\mathbf{S})$ and the actuation commands $b(\mathbf{S})$.

Context Function. The context function h maps the sensor reading vector $\mathbf{S}$ to a natural number representing the context in which an s-bot is, related to the particular task the s-bot is executing. In other words, the context function returns the row index of the activation probability matrix $\mathbf{P}$, thus selecting a set of probabilities used for the definition of the overall s-bot behavior. Therefore, the context function influences the overall behavior of the s-bot changing the activation probabilities of certain basic behaviors. In Sec. 4 we will analyze the context function for the chaining and the clustering task.

Probabilistic Activation Rule. The function p accounts for the probabilistic activation rule, which depends on the context index and the probability matrix. In particular, it returns three behaviors belonging to the three different subsets described above:

$$p : \mathbb{N} \times \mathcal{P} \rightarrow B_m \times B_l \times B_g \tag{4}$$

The selection strategy is the well known *roulette wheel* selection, stating that each basic behavior b has a probability proportional to $P_b(h(\mathbf{S}))$ of being activated at each time step.

Having described the different parts of the probabilistic control architecture, we can summarize the features of this model composing the elements of (1) in the following equation:

$$p(h(s(S)), \mathcal{P})(s(S)) = \langle a_m, a_l, a_g \rangle, \qquad a_m \in A_m,\ a_l \in A_l,\ a_g \in A_g \tag{5}$$

which states that the activation commands $\langle a_m, a_l, a_g \rangle$ are defined by a probabilistic selection of basic behaviors. The behaviors are applied on the reading vector $\mathbf{S} = s(S)$, which is also necessary to define the s-bot context $h(\mathbf{S})$. Furthermore, the probabilistic activation rule relies on the probability activation matrix $\mathbf{P}$, which is defined by the set of parameters $\mathcal{P}$.

It is worth noting that the only elements of (1) that depend on the particular task are the context function h and the set of parameters $\mathcal{P}$. They have to be carefully defined in order to obtain the required overall behavior. As mentioned before, in Sec. 4 we will present these task-dependent part of the control architecture for chain and cluster formations. Except for these elements, the remaining parts are totally independent from the task and allow to generalize the definition of the control for a single s-bot.

4 Pattern Formation

As we have shown above, the behavior of a single s-bot is determined by simple probabilistic choices. Nevertheless, the global pattern arises from the numerous

interactions between individuals. This is a common property in self-organizing systems observable in nature [2]. Examples of pattern formation may be found in physics, chemistry and biology [3]. Chains and clusters are a particular type of patterns, and they are observable in many animal societies. In this section we will describe some examples of chain and cluster formation in natural systems, which are a starting point for the definition of the behavior of a single s-bot and thus of the swarm-bot.

4.1 Chaining Behavior

A chain pattern can be defined as a linear sequence of basic elements, each one connected to at most two other elements. Various species exhibit this pattern, a well-known example being a duck followed by its ducklings. However, it is worth noting that the chaining behavior of ducklings is not an example of self-organization, due to the presence of a leader, the mother duck, which governs the whole group.

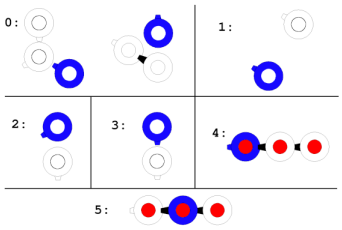
An interesting example of self-assembled chains is given by polymers. *Polymerization* is a chemical process that leads to the creation of long chains of *monomers*, that is, basic elements of a chain. After an initiating reaction with a catalyst reactant, free monomers connect to each other to form chains. These chains attract other free monomers to form longer chains. The process terminates when another reactant molecule connects to the chain, completing the electron octet of the last element. Thus, the individual behavior of a single monomer is a probabilistic connection to other monomers or chains.

The polymer example has many similarities with the formation of chains in the robot swarm. In this case, s-bots are distributed uniformly in the environment and are attracted by other single s-bots to start a chain formation. When an s-bot connects to a chain, it turns on its top light, signaling the presence of a chain. Other free s-bots, being attracted by the light, are stimulated to connect to an existing chain, making the pattern formation more efficient. As in the polymer example, robot chains present physical connections, in order to serve as support, restraint or transmission of mechanical power [8] or to confer to the system elasticity and flexibility properties to better adapt to the environment [10].

As described in Sec. 3, the individual behavior of an s-bot is the result of the probabilistic activation of basic behaviors. The context function $h(s)$ and the probability matrix $\mathbf{P}$ defines the overall behavior of a s-bot. We will now describe these two characterizing parts of the behavioral model. The context function classifies the sensor readings into some relevant situation, pictured in Fig. 2. The corresponding activation probabilities are also shown. At each time step each s-bot evaluates its context and performs some action. The following list explains the actions performed in each possible context:

C0: if an s-bot is near another s-bot or a chain, and does not see a connection point, that is the rear of the other s-bots, repulsion movements have non zero probabilities, in order to let the s-bot go away and search for other connection points. No gripping and no light signaling are performed in this state.

C1: if no s-bot is around, random walk and attraction behaviors have non-zero probabilities, allowing the search for other s-bots or chains. Still no gripping and light signaling are performed.
C2: if an s-bot is nearby and its connection point is reachable, attraction toward it and random movement will try to obtain the right alignment to connect.
C3: if an s-bot sees another s-bot ahead, it closes its gripper with a certain probability $P_c = 1 - P_d$. Here, P_d refers to the probability of disconnection, which influences the chain length distribution (see Sec. 5). The top light is turned on.
C4: if an s-bot is the head of a chain, it switches on its light and does nothing else.
C5: if an s-bot is in the middle of a chain, it turns on the top light and maintain the connection.



	movement					gripper		top light	
	LA	LR	RA	RR	RM	GO	GC	ON	OF
0	0.00	0.10	0.00	0.50	0.40	1.00	0.00	0.00	1.00
1	0.25	0.00	0.35	0.00	0.40	1.00	0.00	0.00	1.00
2	0.00	0.00	0.50	0.00	0.50	1.00	0.00	0.00	1.00
3	0.00	0.00	0.00	1.00	0.00	P_d	$1 - P_d$	1.00	0.00
4	0.00	0.00	0.00	0.00	0.00	1.00	0.00	1.00	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00	0.00

Fig. 2. Perceived contexts and the corresponding activation probability matrix. The null behavior of each subset is not shown: its activation probability can be inferred from the other probabilities and from (3).

It can be noticed that robot chains, different from the polymers, do not grow only from one side, as s-bots can connect to the rear of other s-bot and the head s-bot of a chain is allowed to connect any other s-bot (context **C5**). The simple probabilistic rules and the probability matrix described are sufficient to generate chains: starting from a uniform distribution of s-bots in the environment, it is possible to observe the formation of multiple chains of varying length (see Fig. 3).

4.2 Clustering Behavior

Clusters can be defined as aggregations of objects, and they are observable in many insect species. However, it is not always possible to refer to self-organization. In many cases, individuals react to environmental cues such as illuminance, temperature or humidity, which serve as a template —a particular feature of the environment that influences the pattern formation— for the aggregation process. On the contrary, self-organization in clustering implies some local interaction between individuals, generally in the form of attraction toward other members of the group.

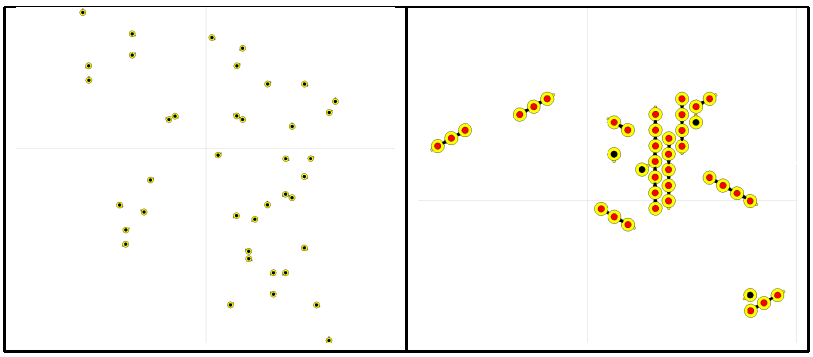


Fig. 3. Chaining: initial distribution of s-bots (left side) and the chain formed after 5000 time steps (right side).

An interesting example of self-organized clustering is observed among bark beetle larvae [3]. These larvae lay pheromone while feeding and react to the presence of pheromone moving in the direction of the gradient of its concentration. The process is auto-catalytic, that is, an increasing density of larvae corresponds to an increasing concentration of pheromone and vice versa. The movement of larvae is given by two components: movement toward zones of higher concentration of pheromone and random walk. Another behavioral component is the *thigmotaxis*, the tendency of maintaining a contact with neighbors. It results that the cluster sizes and distribution deeply depends on the initial density and distribution of larvae.

The clustering behavior of the s-bots is implemented in a way similar to that of the bark beetle larvae. The s-bots are attracted by light and by other s-bots, and turn on their light with a certain probability. Light and s-bot attraction work as the pheromone concentration and thigmotaxis, while the light emission corresponds to the pheromone laying.

In this case, the context function $h(S)$ is related to the number of neighbors felt by a single s-bot. Thus, using local information, an s-bot can evaluate either if it is inside a cluster or in its periphery, and can guess how good the cluster is from the number of neighbors. The more neighbors it has, the better its position in a cluster formation. The activation probabilities of the basic behavior for a given context are listed as a row in the parameter matrix $\mathbf{P}$, shown in Tab. 1. It can be noticed that as the number of neighbors increases, the probabilities of all movement behaviors decrease, in order to let the s-bot search for clusters when alone and stay in formation when clustered. The gripper is never used for clustering, and the top light is switched on with a probability P_l which is used to modulate the cluster size distribution, as it will be shown in Sec. 6.

The overall behavior of the robot swarm results in the formation of multiple clusters of different sizes. The initial disposition of s-bots in the environment

Table 1. Activation probability matrix for the clustering behavior

	movement					gripper		top light	
State	LA	LR	RA	RR	RM	GO	GC	ON	OF
S0	0.30	0.00	0.20	0.00	0.50	0.00	0.00	P_l	$1 - P_l$
S1	0.20	0.00	0.20	0.00	0.50	0.00	0.00	P_l	$1 - P_l$
S2	0.15	0.00	0.15	0.00	0.50	0.00	0.00	P_l	$1 - P_l$
S3	0.10	0.00	0.10	0.00	0.30	0.00	0.00	P_l	$1 - P_l$
S4	0.10	0.00	0.05	0.00	0.10	0.00	0.00	P_l	$1 - P_l$
S5	0.10	0.00	0.05	0.00	0.05	0.00	0.00	P_l	$1 - P_l$
S6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	P_l	$1 - P_l$

is chosen randomly and uniformly. A snapshot of a formed cluster is shown in Fig. 4.

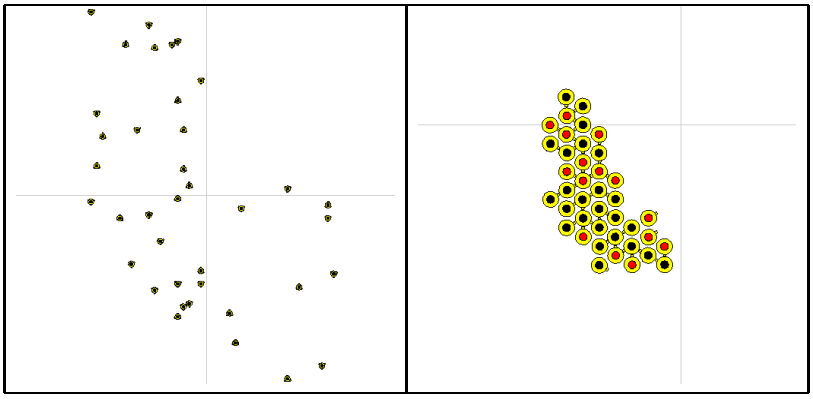


Fig. 4. Clustering: initial distribution of s-bots (left side) and final cluster (right side).

5 Chain Formation

In this section we analyze the chaining behavior described in Sec. 4.1, in order to understand the dynamics that leads to a certain distribution of chains. In particular, we are interested in studying the mean chain length in relation to the probability of disconnection P_d. For this purpose, we have developed an analytical model that can predict the distribution of chain lengths, which will be presented in Sec. 5.1. The parameters of this model can be estimated using the simulated experiences. The estimation procedure will be described in Sec. 5.2. Finally, in Sec. 5.3 the predictions of the mathematical model are compared with the results of the simulations.

5.1 Mathematical Model

The mathematical model used to describe the chaining behavior is similar to those used in modeling polymerization processes. Given N robots, $X_i(t)$ refers to the average number of chains of length i (called i-chain) at time t, where $i \in [1, N]$. Thus, we have N variables, whose evolution in time can be described by a set of N differential equations. The time dependence of the variables is omitted to simplify the notation.

$$\dot{X}_1 = -\sum_{i=1}^{N-1} k_i X_1 X_i + \sum_{i=2}^{N} e_i X_i - k_1 X_1^2 + e_2 X_2 \tag{6}$$

$$\dot{X}_i = k_{i-1} X_1 X_{i-1} + e_{i+1} X_{i+1} - k_i X_1 X_i - e_i X_i \tag{7}$$

This model explains how chains grow and shorten according to given connection and disconnection rates. In general, these parameters vary over time and are depending on the chain length. The connection rates are represented as an N-dimensional vector $\mathbf{K}$, each element $k_i = k_i(t)$ referring to the rate of connection of a robot to a i-chain. Similarly, the disconnection rates are represented by the vector $\mathbf{E}$, and the rate of disconnection from a i-chain is given by its element $e_i = e_i(t)$.

Equation (6) accounts for the variation of the number of chains of unit length, that is, free robots: they can connect to other chains (the first term in the right side of (6)), thus decreasing X_1 which is increased by other robots disconnecting from other chains (second term). The third and fourth terms takes into account the formation and disband of 2-chains, which both involve an additional robot. Equation (7) accounts for the number of i-chains: the increase in the number is due to the growth of shorter chains or to the shortening of longer chains as represented by the first and second term in (7). The decrease is due to the connections/disconnection of free robots to/from them, accounted by the last two terms in (7). The upper limit condition is given by $X_{N+1} = 0$, as it is not possible to have a chain of length $N + 1$. Furthermore, the analytical model has the following properties:

$$\sum_{i=1}^{N} i X_i = N, \qquad \sum_{i=1}^{N} i \dot{X}_i = 0 \tag{8}$$

which states that the number of robots must be constant during time. In the following section we describe the parameter estimation procedure from the simulations of the swarm of robots.

5.2 Parameter Estimation

The described analytical model needs the parameter vector $\mathbf{K}$ to be estimated, in order to have a strict correspondence between the model and the simulations. Estimating these parameters is not trivial. Often, the estimation procedures are

limited by the particular system being studied: for example, in polymerization, it is possible to have quantitative evaluations of some parameters by means of the observation of the system as a whole. In our case, as the observed system is simulated, we have a full access to it and to each individual state. Nevertheless, changes to the control of the s-bots or to the system have to be avoided or carefully designed, as they may introduce additional errors in the estimation procedure.

We will now introduce the estimation procedure for a given parameter k_i, which corresponds to the connection rate of a single s-bot to a i-chain. The connection rate can be roughly defined as the reciprocal of the average time an s-bot needs to connect to an i-chain. Starting from this definition, we conducted systematic experiments to estimate this value. Initially, 160 s-bots are placed in an arena of 42×42 cells. This makes the initial spatial density close to the one observed in the simulations. Robots are either free or part of an i-chain. Thus, among the 160 s-bots, $\hat{X}_1$ are free robots while the rest is placed in $\hat{X}_i$ i-chains and cannot move at all. These chains can be seen as "seeds" to which a free s-bot can connect. The initial disposition of free robots and chains is randomly uniform. We measure the time T_c until the first connection between a free s-bot and a seed. Under the assumption that $\hat{X}_1$ and $\hat{X}_i$ do not change during a single experiment, k_i is estimated by:

$$\hat{k}_i = \frac{1}{\langle T_c \rangle \ \hat{X}_1 \ \hat{X}_i}, \tag{9}$$

where $\langle T_c \rangle$ is the average time on 100 runs. The presence of $\hat{X}_1$ and $\hat{X}_i$ are necessary because their value influences T_c: the more the free s-bots and seed, the shorter the time until the first connection. It could be possible that free s-bots connect to each other, thus changing $\hat{X}_1$ during time. To overcome this, the value used for $\hat{X}_1$ is high enough to assure that, if any connection happens, the variation is proportionally small. Moreover the behavior of free robots after a connection was slightly changed in such a way that they ungrip immediately if they connect to other s-bots. In this way, the formation of small chains between them is prevented, because it could interfere with the estimation of k_i. Disconnection rates correspond to the disconnection probabilities P_d introduced in Sec. 4.1, thus they do not have to be estimated. The result of this procedure is shown in Fig. 5. It can be noticed that the connection rates to free s-bots are nearly six times those for a 2-chain. This suggests that the connection point to a chain is more difficult to be reached by an s-bot, as the chain does not move. Nevertheless, longer chain presents a slight increase in the connection rate, which may be related to the bigger attraction a longer chain has on free s-bots. Further insights in the parameter estimation results are given in the next section, under the light of the comparison between the simulated system and the analytical model.

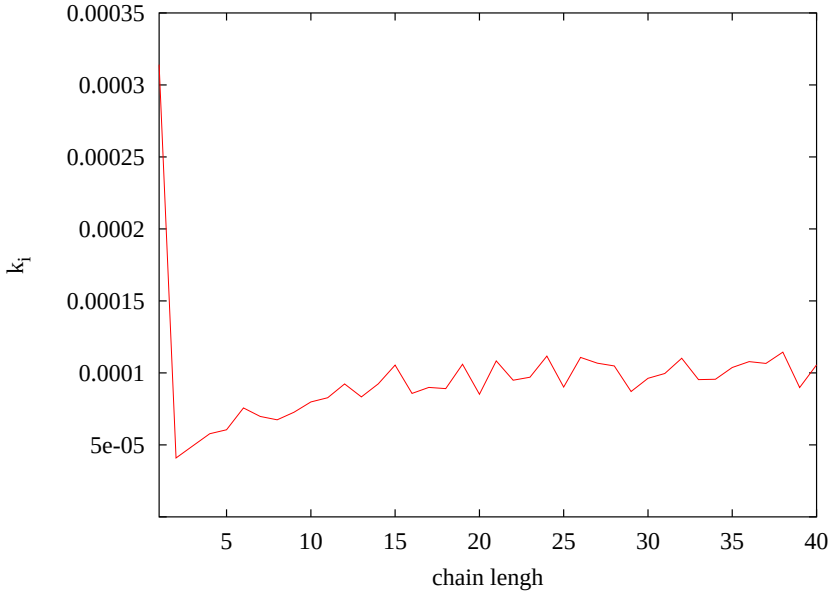


Fig. 5. Estimated k_i values

5.3 Results

We present here the results obtained with the analytical model and the corresponding simulations. Fig. 6 shows the distribution of a simulated population of 40 s-bots for different disconnection probabilities, namely $P_d = 0.01$ and $P_d = 0.07$. The values in the graphs represent the average number of s-bots that have been observed in chains of different length and is evaluated by the following expression:

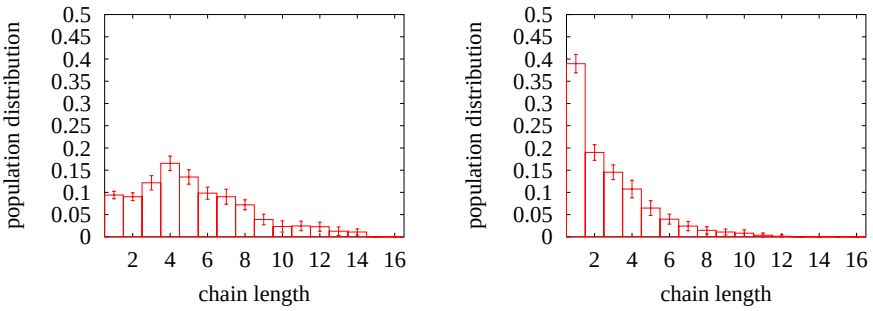
$$\frac{iX_i}{\sum_{j=1}^{N} jX_j} = \frac{1}{N}\, iX_i \tag{10}$$

where N is the total number of s-bots and X_i is the mean number of chains of length i. These graphs describe well the effect of increasing the disconnection probability: bigger values prevent long chains to appear.

In Fig. 7 we plot the mean length of chains to which in average an s-bot belongs, depending on disconnection probabilities in the range $[0, 0.1]$ with increment of 0.01. These values are evaluated by the following equation:

$$\frac{\sum_{i=1}^{N} i^2 X_i}{\sum_{i=1}^{N} iX_i} = \frac{1}{N} \sum_{i=1}^{N} i^2 X_i. \tag{11}$$

For $P_d = 0$, s-bots belong in average to chains that are three or four units long. When increasing P_d, first there is a little increase of this value, followed



(a) $P_d = 0.01$ (b) $P_d = 0.07$

Fig. 6. Population distribution observed from the experiments evaluated during time-step 2001 until 3000. Average on thirty experiments is plotted. Bars represent standard deviations. The y axis is normalized to the number of robots

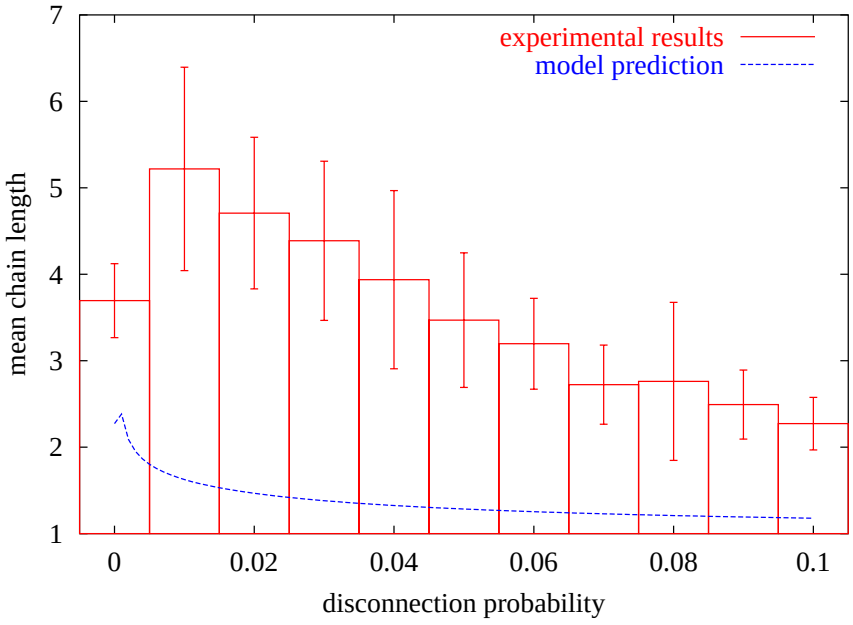


Fig. 7. Comparison between experimental results and the estimated values for k_i.

then by a strictly monotonic decrease. In the same figure, the analytical model prediction obtained with the estimated k_i is shown. The disagreement between the curves is considerable.

In order to understand why this big discrepancy arose, we tried to find out a better matching between the experimental data and the analytical model, under the hypothesis of a given dependency between the connection rates k_i and the chain length i. We have hypothesized four different situations:

- $k_i = \alpha \cdot \hat{k}_i$, where an under-estimation of the connection is envisioned.
- $k_i = \alpha$, where different length does not influence the connection rates.
- $k_i = \alpha \cdot i$, where longer chains attract more s-bots than shorter ones.
- $k_i = \frac{\alpha}{i}$, where longer chains have lower connection rates.

The parameter α was hand-tuned in all cases. At this stage, we are interested only in a qualitative comparison, thus no statistical test but only a superimposition of different curves was performed, in order to observe which one shows a better fit. The corresponding results are shown in Fig. 8.

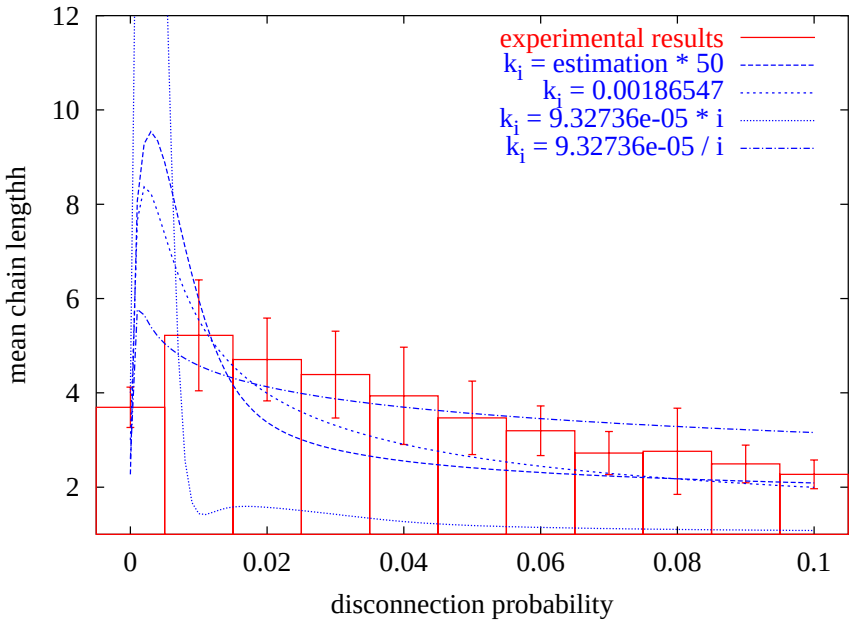


Fig. 8. Comparison between different functions for k_i and experimental results. All functions are plotted in the range $[1, 12]$, thus truncating $k_i = i \cdot 9.3\ 10^{-5}$, to see better the differences.

The under-estimation hypothesis ($k_i = \alpha \cdot \hat{k}_i$) has been tested with $\alpha = 50$. It fits the experimental data satisfactorily for low and high values of P_d, but not

in between. The constant value hypothesis ($k_i = \alpha$) results in a better fitting for low P_d, but it still presents a different behavior in the range $[0.03, 0.06]$. The proportional hypothesis ($k_i = \alpha \cdot i$) is justified recalling that s-bots are attracted toward light sources and longer chains creates a bigger attraction field with respect to shorter ones. The obtained distribution results in a mean length around 1 nearly for all P_d, except for a peak of value 21 for $P_d \approx 0.003$. Finally, the inversely-proportional hypothesis $k_i = \frac{\alpha}{i}$ can be justified observing that a longer chain attracts robots in average near its center, where there are no possibilities of connections. Thus, an s-bot has to cover more space to reach the rear or the front of the chain. This hypothesis gave better results than the others, fitting well in $[0, 0.6]$ but overestimating after this values.

None of the above functions fit completely the experimental data. Thus, it is necessary to understand if the model and the parameter estimation process are correct.

In general, a model is based on some assumptions and approximations that make it not completely adherent to the modeled system. These simplifications aim to identify some main features of the system, resulting in an easier and faster tool used to make predictions on the system behavior.

Among these simplifications, the developed model does not take into account any spatial information, which is related to robot-robot interactions. In fact, in the model it is assumed that each robot has probability k_i to connect to an i-chain in a time unit, discarding their relative positions. But robots are attracted by light sources and other robots, thus their connection probability depends on chain densities in the neighborhood of the robot. We will return to this point in Sec. 6.3, referring to cluster formation.

The simplification used in the model can still be considered a good approximation in case s-bots and chains have, in average, a uniform spatial density. This is not the case in the chaining behavior, where the first phase consists essentially in a clustering behavior. Moreover, each formed chain creates an attraction field for other s-bots from which it is hard to escape once caught in. This implies that each chain has a cloud of robots around it, preventing them to perform a free random walk all over the arena. A quantitative insight of this effect is given in Fig. 9: we computed the minimum rectangular area that contains all the robots during a normal run of our simulation. This measure gives us an estimation of the area occupied by the robots, and it is inversely proportional to the mean spatial density. Fig. 9 shows the results for three disconnection probabilities. The ratio between the final and the initial areas is 0.43 for $P_d = 0.01$, 0.20 for $P_d = 0.05$ and 0.15 for $P_d = 0.1$, confirming a sensible increase in spatial density during time.

The variation in spatial density may influence also the parameter estimation process. We have estimated the connection rates k_i having an initial spatial density close to the one observed in the simulations, and we kept it constant. However, as the spatial density changes over time, this may influence the connection rates. Moreover, the disconnection probability determines also the amount of change in spatial density, suggesting that it can influence the k_i as well. Thus,

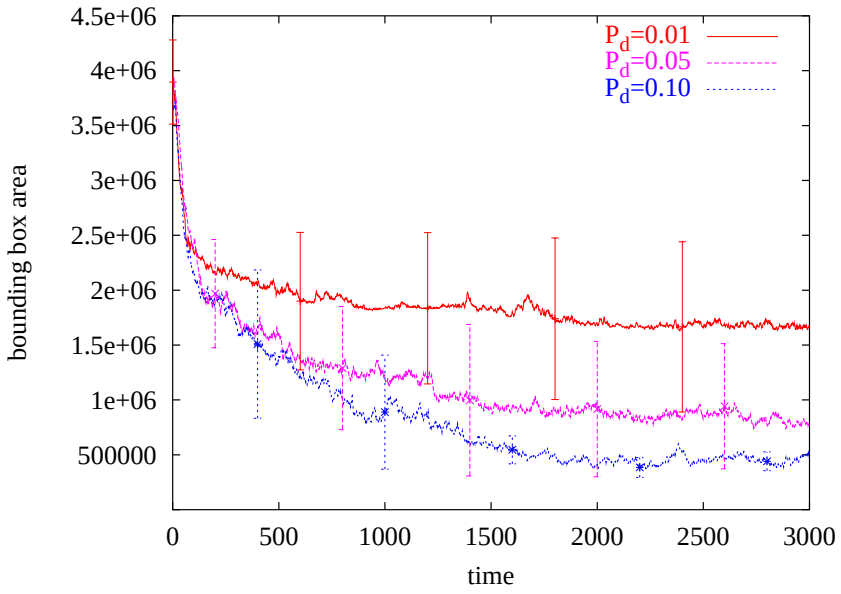


Fig. 9. Area occupied by the robots during simulations for chain formation. The mean over ten runs is plotted. Bars represent standard deviations.

the connection rates seem to depend on both time and the disconnection probability.

These considerations suggest that, if we assume the correctness of the model, a better experimental setup for the estimation of k_i is required, in order to take into account dependencies on time and on P_d.

6 Cluster Formation

In this section, we describe how the cluster formation is affected by some parameters of the s-bot behavior. Similar to the chain pattern, we are interested in the distribution of cluster sizes as a function of the probability of switching on/off the s-bot light. For this purpose, we present in Sec. 6.1 an extension of the model used for chaining. Section 6.2 presents the modifications needed in the parameter estimation process with respect to the procedure described in Sec. 5.2. Finally, in Sec. 6.3 we discuss the obtained results.

6.1 Mathematical Model

Given N robots, $X_i(t)$ refers to the average number of clusters of dimension i (called i-clusters) at time t. A system of N differential equations is proposed to

describe the dynamics of the X_i:

$$\begin{aligned}\dot{X}_1 = & -\sum_{j=2}^{N} j k_{1,j-1} X_1^2 + \sum_{i=2}^{N} i e_{i,i-1} X_i \\ & -\sum_{i=2}^{N-1} \sum_{j=1}^{N-i} j k_{i,j} X_1 X_i + \sum_{i=3}^{N} \sum_{j=1}^{i-2} j e_{i,j} X_i, \end{aligned} \tag{12}$$

$$\begin{aligned}\dot{X}_i = & -\sum_{j=1}^{N-i} k_{1,j} X_1 X_i - \sum_{j=1}^{i-1} e_{i,j} X_i \\ & +\sum_{j=1}^{i-1} k_{i-j,j} X_1 X_{i-j} + \sum_{j=i+1}^{N} e_{j,j-i} X_j, \end{aligned} \tag{13}$$

where $k_{i,j} = k_{i,j}(t)$ is the connection rate of j robots to a i-cluster, and, similarly, $e_{i,j} = e_{i,j}(t)$ is the disconnection rate of j robots from a i-cluster. The first row in (12) accounts for the variation of the number of free robots due to two effects: the formation of a j-cluster from free robots (first term) and the complete disband of a i-cluster. The second row accounts for two different events: the growth of a i-cluster of j robots (first term) and the disconnection of j robots from a i-cluster. In (13) the first row describes the decrease of X_i due to the growth of an i-cluster (first term) and for the disconnection of j robots from an i-cluster (second term). The second row describes the increase of X_i due to the growth of a $(i-j)$-cluster when j robots connect to it (first term) and the disconnection of $(j-i)$ robots from a j-cluster (second term). Even in this case, the upper limit condition is simply given by $X_{N+1} = 0$.

This model is clearly an extension of the model used for chaining, allowing multiple connections per time step. In fact, if we substitute in (12) and (13) $k_{i,j} = 0$, $e_{i,j} = 0$ for all $j > 1$, that is, we impose that no more than one robot per time step can connect or disconnect, then we obtain (6) and (7). The parameters of the model can be collected in two matrices, $\mathbf{K}$ and $\mathbf{E}$. The estimation process that defines this parameters is explained in the following section.

6.2 Parameter Estimation

The parameter estimation process for clustering is aimed to define the $\mathbf{K}$ and $\mathbf{E}$ matrices from systematic experiments. The faster dynamics of the clustering process and the higher number of parameters to be estimated make this task more complex than in the case of chains formation. The procedure described in Sec. 5.2 is adapted distributing free robots and cluster seeds randomly in the environment and observing the first connection to and disconnection from a seed. This methodology leads to the following estimators:

$$\hat{k}_{i,1} = \frac{1}{\langle \frac{T_c}{n_c} \rangle \hat{X}_1 \hat{X}_i} \tag{14}$$

$$\hat{e}_{i,1} = \frac{1}{\langle \frac{T_d}{n_d} \rangle \hat{X}_i}, \tag{15}$$

where T_c (T_d) is the time elapsed until the first connection (disconnection) of a single s-bot to (from) a i-cluster, and n_c (n_d) is the number of contemporary connections (disconnections). In this case, contemporary connections or disconnections appear very often, thus it is necessary to consider them in the estimation. However, this methodology has drawbacks for the clustering case,

ranging from the complexity of estimating the $k_{i,j}$ and the $e_{i,j}$ for $j > 1$ to problems given by the fast dynamic of the clustering process, which implies a non-negligible dependency from the starting condition and an high variance of the results, above all for cluster seeds of small size.

A different parameter estimation methodology has been proposed, in order to overcome the above described problems. In this methodology, a single i-cluster seed of non-moving s-bots is placed in the environment. Initially connected to this seed, an additional s-bot is free to move following its control rules, thus connecting and disconnecting from the seed. During time, we compute the connection and disconnection rates of the free s-bot as described by the following equations:

$$\hat{k}_{i,1}(t) = \frac{n_c(t)}{T_d(t)} \tag{16}$$

$$\hat{e}_{i+1,1}(t) = \frac{n_d(t)}{T_c(t)}, \tag{17}$$

where $n_c(t)$ ($n_d(t)$) is the number of connections (disconnections) observed until time t, and $T_d(t)$ ($T_c(t)$) is the time spent by the s-bot unconnected from (connected to) the i-cluster. In other words, $T_d(t)$ ($T_c(t)$) counts the number of possibilities that the s-bot has to connect (disconnect), and $n_c(t)$ ($n_d(t)$) counts the number of possibilities that are effectively exploited. The so computed values vary over time and usually reach a stable state after few thousand of time steps. Thus, the estimators are finally given by:

$$\hat{k}_{i,1} = \langle \hat{k}_{i,1}(T) \rangle \tag{18}$$

$$\hat{e}_{i+1,1} = \langle \hat{e}_{i+1,1}(T) \rangle, \tag{19}$$

where the average is made over 20 experiments and $T = 10000$ is the time step at which most of the experiments have shown a reached stability of $\hat{k}_{i,1}(t)$ and $\hat{e}_{i+1,1}(t)$. Figure 10 shows the estimated connection and isconnection rates for a probability of switching on the light $P_l = 0.30$, along with their standard deviation. It can be noticed that the connection rate increases with the cluster size, while the disconnection rate decreases. The standard deviation in connection rates increases with the cluster size, suggesting that the experiments have a high variability, due to the difficulty of disconnection from the cluster, hypothesis confirmed by the disconnection rate nearly equal to zero.

The same methodology can be applied to the estimation of $k_{i,j}$ and $e_{i,j}$, given $j > 1$. It is sufficient to place j moving s-bots connected to a given i-cluster and measure the following quantities:

$$\hat{k}_{i,j}(t) = \frac{n_c^j(t)}{T_d^j(t)} \tag{20}$$

$$\hat{e}_{i+j,j}(t) = \frac{n_d^j(t)}{T_c^j t} \tag{21}$$

where $n_c^j(t)$ ($n_d^j(t)$) is the number of contemporary connections (disconnections) of j s-bots observed until time t, and $T_d^j(t)$ ($T_c^j(t)$) is the time spent by all

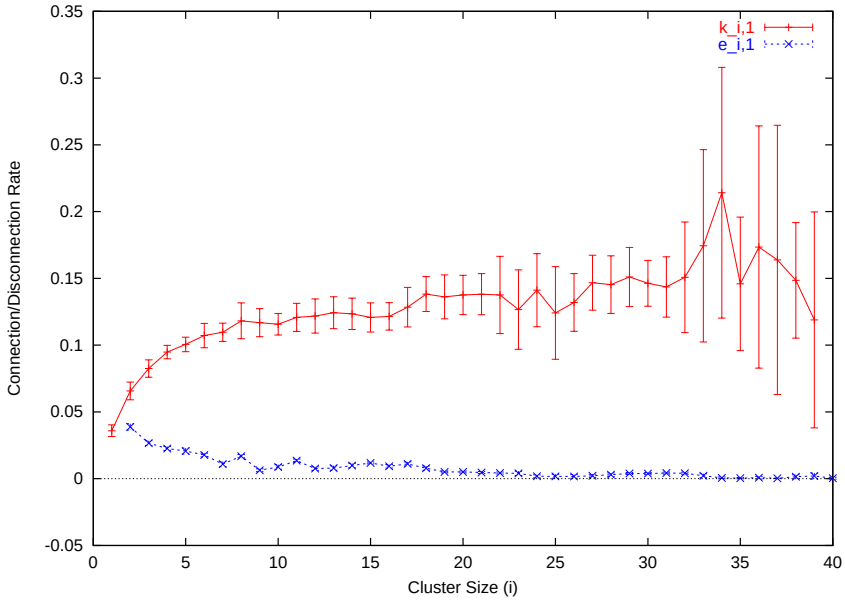


Fig. 10. Connections and disconnection rates for a single robot to a i dimension cluster ($\hat{k}_{i,1}(t)$ and $\hat{e}_{i,1}(t)$), estimated in experiments with probability of switching on the light $P_l = 0.30$

j s-bots unconnected from (connected to) the i-cluster. The estimator can be obtained extending in the same direction (18) and (19).

This procedure can be criticized because of its limited fitting with the experimental situation, where it is not always present a single cluster. However, when the cluster formation process reaches its stationary state, the different clusters that have formed are enough distant from each other to not interfere, making possible to simplify the analysis to a single cluster. This consideration justifies the applied estimation procedure.

6.3 Results

The analytical model parameters are estimated for different probabilities P_l of switching on the light. In particular, we analyzed two probability values, corresponding to different stationary states: $P_l = 0.03$ always leads to the formation of a single cluster, while $P_l = 0.30$ results in multiple clusters of different sizes. The data are extracted simulating a swarm of 40 s-bots randomly positioned at the beginning of each experiment in a square area of 21 cells side. We measured the cluster sizes at the end of 100 experiments, in order to evaluate the distribution of the s-bot population over the different cluster sizes. This measure is exactly the same one described in (10), which defines the percentage of s-bots

involved in a cluster structure of size i. Averaging the values obtained in the different experiments, we obtain the distribution shown in Fig. 11.

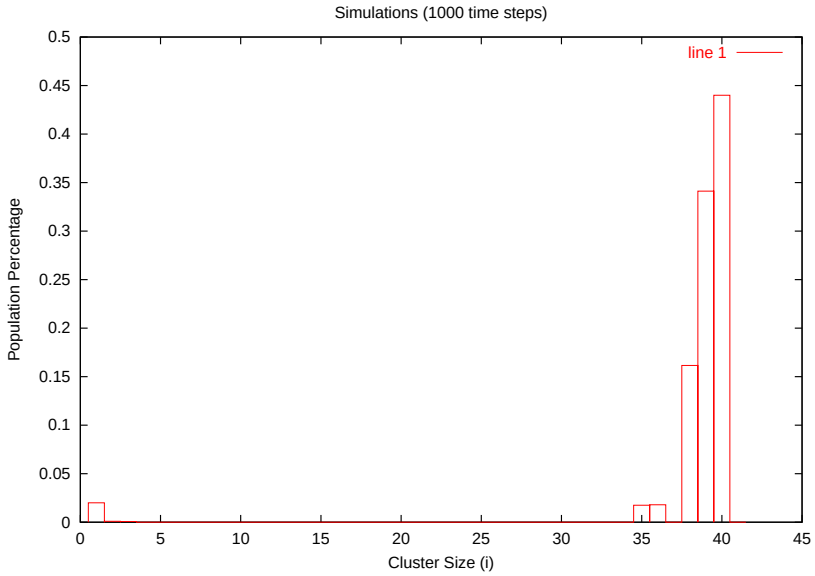


Fig. 11. Cluster distribution obtained from simulations having $P_l = 0.03$.

It is worth noting that this distribution corresponds to a single cluster involving all the s-bots (see also Fig. 4). However, a non zero percentage of robots statistically belong to slightly smaller clusters, and free robots are also present. This is explained observing that the cluster structure is dynamic and many s-bots can disconnect and connect in each time step, leading to the observation of different cluster sizes.

A similar behavior is predicted by the analytical model, where the same measures have been computed. The analytical model has been numerically integrated starting from the condition $X_1 = N$, which represent an initial population of free robots. The obtained results after 1000 time steps are shown in Fig. 12. It can be noticed that the distribution of the robot population is qualitatively similar. Nevertheless, the analytical model has not reached the stationary state, as some small clusters are still present.

The stationary state is reached after 100000 time steps, as can be observed in Fig. 13. In this case, the analytical model and the simulations show results quantitatively similar.

Similar experiments have been carried out for probability of switching on the light $P_l = 0.30$. In this case s-bots keep the top light on in average for more time, resulting in a stronger attraction between individuals. Thus, we expect that

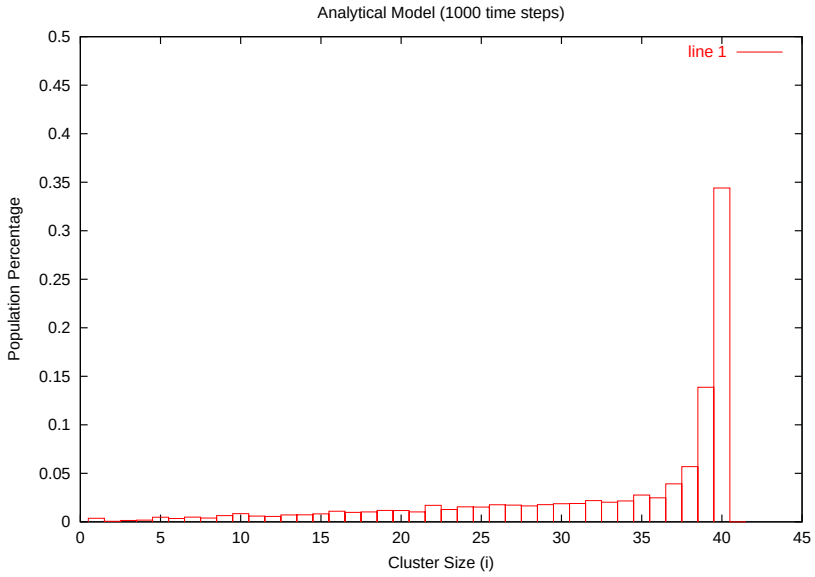


Fig. 12. Cluster distribution obtained from the mathematical model after 1000 time steps, having $P_l = 0.03$

the number of small clusters increases, because even a small cluster creates an attraction field strong enough to trap s-bots in its neighborhood. This intuition is confirmed by the experimental results obtained from simulation, shown in Fig. 14. It can be noticed that, in average, s-bots belong to medium-size clusters.

The analytical model has been used also in this case, and the corresponding results, obtained after 1000 time steps, are shown in Fig. 15. It can be noticed that results are qualitatively similar. In fact, the analytical model correctly predicted the coexistence of medium-size clusters. One important difference is in the number of clusters of dimension 40, the maximum size. The simulation suggests that in some cases a single cluster may arise, but the analytical model predicts that the appearance of such a formation is rare. Thus, as in the case of $P_l = 0.03$, we can suppose that the analytical model has not reached the stationary state after 1000 time steps.

The analysis of the theoretical distribution in the stationary state is given in Fig. 16, where the mathematical model is simulated for 100000 time steps. Evidently, they are different from what we expected: the analytical model converges to a single-cluster solution, even more stable than in the case of $P_l = 0.03$ as the values of the smaller clusters are negligible.

These results are useful to explain an important feature of the introduced model that is not fully predictable: the analytical model does not take into account the spatial information, which we supposed being embedded in the estimated parameters $\mathbf{K}$ and $\mathbf{E}$. It can be observed that the way in which we

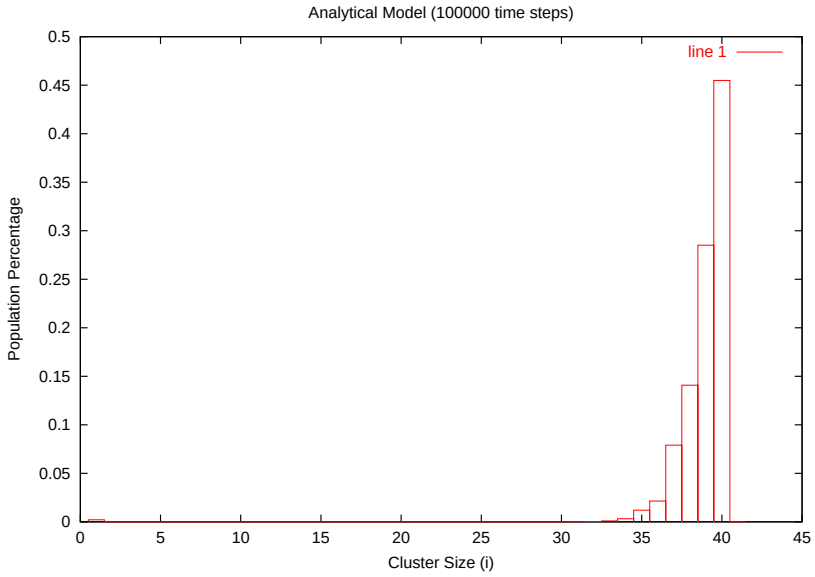


Fig. 13. Cluster distribution obtained from the mathematical model after 100000 time steps, having $P_l = 0.03$.

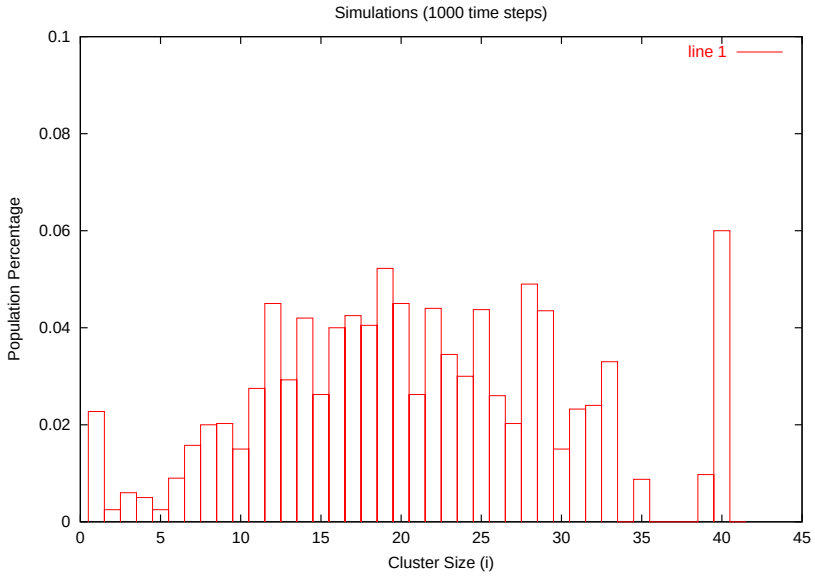


Fig. 14. Cluster distribution obtained from simulations having $P_l = 0.30$.

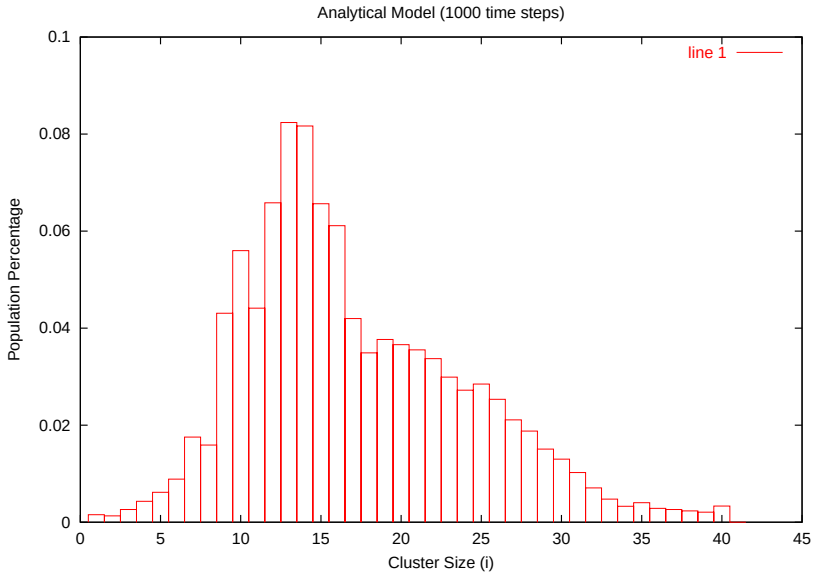


Fig. 15. Results from the analytical modeling of cluster formation after 1000 time steps, with $P_l = 0.30$.

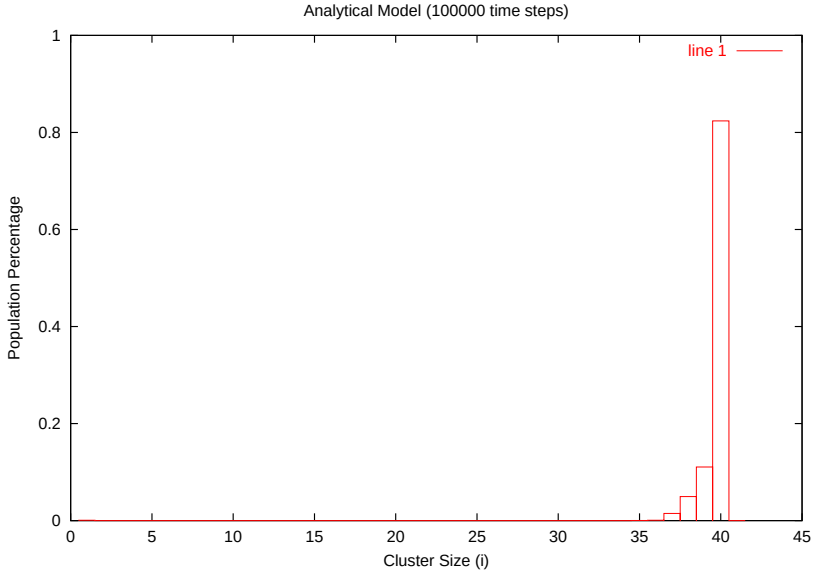


Fig. 16. Results from the analytical modeling of cluster formation after 100000 time steps, with $P_l = 0.30$.

estimated the parameters lacks of the required information about a spatial distribution. However, a deeper insight in the analytical model show us the following property: every time an s-bot disconnects from an i-cluster, it is referred in the model with a decrease in the average number of i-cluster X_i and an increase of the average number of free robots X_1. Thus, X_1 represents a pool from which every X_i can gather robots. This behavior of the model does not represent in any case the simulated situation in which, when a s-bot disconnect from a i-cluster, it reconnect more likely to the original cluster than to another one, due to the higher attraction from the original cluster.

7 Discussion

The process of self-assembly is complex and any attempt to design controllers for individuals to self-assemble into a desired pattern will be greatly helped by a better understanding of the system as a whole. The proposed methodology is aimed to complement the design of the controllers with analytical modeling of the system.

Figure 17 depicts our approach by putting the studies presented in the previous sections into a framework. The upper left box represents the actual system for which the control algorithms are developed. Within the context of this report, the system is the simulation of a swarm of s-bots controlled by the described architecture (see Sec. 3).

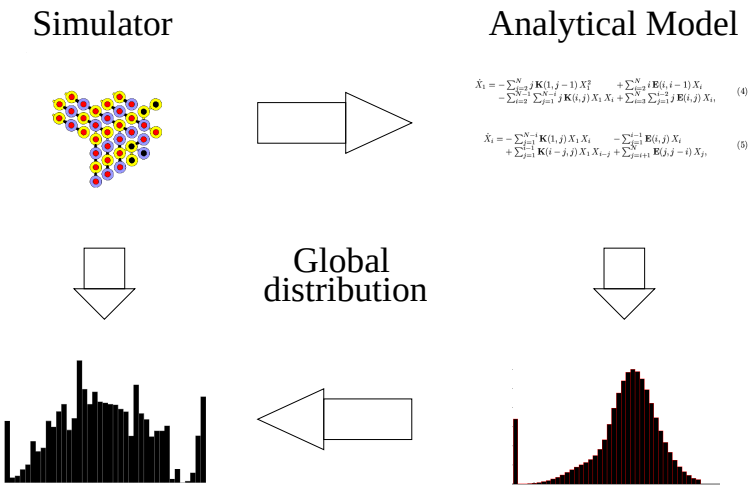


Fig. 17. Experiment-Theory

An analytical model of the system, depicted as the upper right box, is defined. The model is parametrized in such a way that the parameters are measurable

from the actual system through systematic experiments. For instance in chaining, the connection probability of single s-bots to chains of different lengths constituted such parameters. These parameters are then substituted in the analytical model to make predictions about the global behavior of the actual system. In chaining, for instance, the distribution of chain length is one of the global observations, that can be used to compare with the one obtained from the analytical model. This comparison provides a cross-check for the analytical model, stating that the model and the parameters measured are sufficient to model the behavior of the actual system.

This cross-check, by itself, does not improve our understanding of the system. However by confirming its equality to the actual system, it paves the way for us to claim that other analyses made with the analytical model are likely to be transferable to the actual system.

In our project, differently from similar studies in chemistry and biology, the goal is to use the analysis obtained from the analytical model as a guide to the design of the s-bots control model. For instance, if the analytical model shows that one of the control parameters of the model needs to be within a certain range to obtain a particular global behavior, then we can concentrate on changing the control to move the measured value of that parameter to that particular range. Although we do not claim that this methodology will provide us with such ability, we believe that it will guide us in the right way.

However, the obtained results are still not satisfactory. The correctness of the developed model is confirmed by the observed dynamics, that are very similar to the real case in both chaining and clustering. Nevertheless, in chaining the stationary state reached by the model can be compared only qualitatively with the actual results, and in clustering this does not happen at all. As we mentioned in Sec. 6.3, it is likely that the developed model lacks of spatial information, which plays a major role in defining the stationary state. This information must be coded explicitly or implicitly in the model: explicitly, adding the dependence from space or time to the connection and disconnection rates, dependency that has been neglected till now; implicitly, improving the parameter estimation process in such a way that make it responsible for the missing information, under the assumption that these parameters can be considered constant in time.

Furthermore, some other discrepancies between the analytical model and the swarm simulations must be taken into account:

- the simulations are carried out in discrete time, while the analytical model is integrated in continuous time. In order to have a better comparison of the predicted and actual dynamics, it is required coherence between timings that is currently missing;
- it seems that the long range interaction between s-bots is not taken into account by the developed analytical model. This kind of interaction is also related to spatial information, making an s-bot able to sense other robots at very high distances. This kind of interaction is not present in the natural systems that we take as inspiration, and therefore is not accounted by the analytical model.

Instead of updating the model in order to resolve these problems, we believe that it is more important to have more realistic simulations. Working in a continuous world and having a more realistic sensing will decrease or completely cancel the above discrepancies between the analytical model and the simulations.

8 Conclusions

In this report, we presented a behavioral architecture for a swarm of robots which has been demonstrated very effective for self-assembling tasks. This architecture is independent from the particular task the robot have to accomplish, except for the context function and the parameters matrix. Future developments can follow two different directions: from one side, as the definition of the parameter matrix is complex, genetic algorithms may be applied to evolve the parameters, using the context function as a fitness function. On the other side, the architecture can be modified in order to let the context function directly update a single set of activation probabilities and make the system auto-adaptable to the specified task. This solution is more similar to a reinforcement learning of the activation probabilities, the context function playing the role of the reward function.

We also presented a framework for analyzing the self-assembling process using an analytical model. Although the obtained results are still not satisfactory, we believe that a similar approach can help the designer to have a better understanding of the dynamics of the self-assembling process. This approach can be used either to explain the process dynamics from a global point of view or to make predictions on the parameters that govern it. Thus, future directions concerns also a refinement of this methodology, along with an improvement of the simulated environment toward more realistic simulations.

References

[1] C. Anderson, G. Theraulaz, and J.L. Deneubourg. Self-assemblages in insect societies - review article. *Insectes Sociaux*, 49:1–12, 2002.

[2] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, NY, USA, 1999.

[3] S. Camazine, J.L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organisation in Biological Systems*. Princeton University Press, NJ, USA, 2001.

[4] E. Şahin, T. H. Labella, V. Trianni, J.-L. Deneubourg, P. Rasse, D. Floreano, L. M. Gambardella, F. Mondada, S. Nolfi, and M. Dorigo. Swarm–bot: Pattern formation in a swarm of self–assembling mobile robots. submitted to SMC-2002, International Conference on System, Man and Cybernetics.

[5] O. Holland and C. Melhuish. Stigmergy, self-organization and sorting in collective robotics. *Artificial Life*, 5(2), 2000.

[6] C. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30:85–101, 2000.

[7] K. Lerman, A. Galsyan, A. Martinoli, and A. Ijspeert. A macroscopic analytical model of collaboration in distributed robotic systems. *Artificial Life*, 7:375–393, 2001.

[8] F. Mondada, A. Guignard, A. Colot, D. Floreano, J.-L. Deneubourg, L. M. Gambardella, S. Nolfi, and M. Dorigo. SWARM-BOT: A new concept of robust all-terrain mobile robotic system. submitted to 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems.

[9] A. Pamecha, C. Chiang, D. Stein, and G. Chirikjian. Design and implementation of metamorphic robots. In *Proceeding of the ASME Design Engineering Technical Conference and Computers in Engineering Conference*, Irvine, CA, USA, 1996.

[10] S. Rasmussen, N.A. Baas, B. Mayer, M. Nilsson, and M.W. Olesen. Ansatz for dynamical hierarchies. *Artificial Life*, 7:329–353, 2001.

[11] M. Yim, Y. Zhang, J. Lamping, and E. Mao. Distributed control for 3D metamorphosis. *Autonomous Robots*, 10:41–56, 2001.