

---

# Introduction

---

## 1.1 SOCIAL INSECTS

Insects that live in colonies, ants, bees, wasps,<sup>1</sup> and termites, have fascinated naturalists as well as poets for many years. “What is it that governs here? What is it that issues orders, foresees the future, elaborates plans, and preserves equilibrium?,” wrote Maeterlinck [230]. These, indeed, are puzzling questions. Every single insect in a social insect colony seems to have its own agenda, and yet an insect colony looks so organized. The seamless integration of all individual activities does not seem to require any supervisor.

For example, Leafcutter ants (*Atta*) cut leaves from plants and trees to grow fungi (Figure 1.1). Workers forage for leaves hundreds of meters away from the nest, literally organizing highways to and from their foraging sites [174].

Weaver ant (*Oecophylla*) workers form chains of their own bodies, allowing them to cross wide gaps and pull stiff leaf edges together to form a nest (Figure 1.2). Several chains can join to form a bigger one over which workers run back and forth. Such chains create enough force to pull leaf edges together. When the leaves are in place, the ants connect both edges with a continuous thread of silk emitted by a mature larva held by a worker (Figure 1.3) [172, 174].

---

<sup>1</sup>Only a fraction of all species of bees and wasps are social: most are solitary.

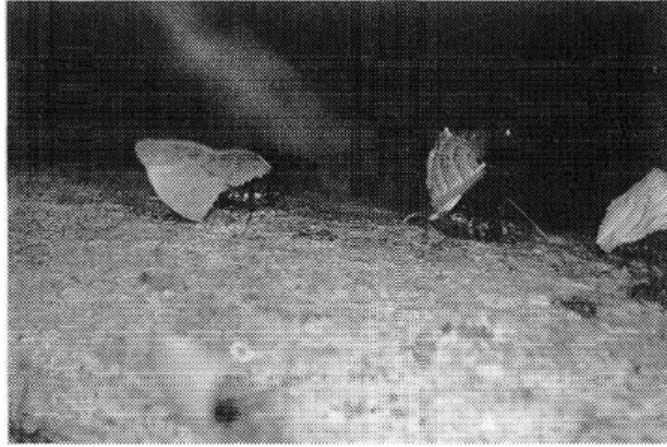


FIGURE 1.1 Leafcutter ants (*Atta*) bringing back cut leaves to the nest.

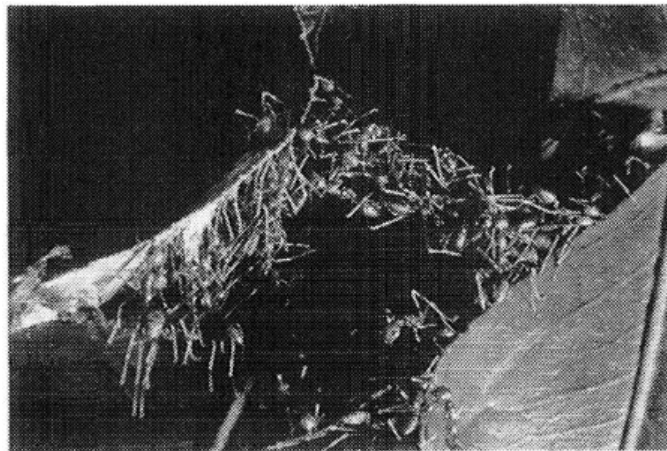


FIGURE 1.2 Chains of *Oecophylla longinoda*.

In their moving phase, army ants (such as *Eciton*) organize impressive hunting raids, involving up to 200,000 workers, during which they collect thousands of prey (see chapter 2, section 2.2.3) [52, 269, 282].

In a social insect colony, a worker usually does not perform all tasks, but rather specializes in a set of tasks, according to its morphology, age, or chance. This division of labor among nestmates, whereby different activities are performed simultaneously by groups of specialized individuals, is believed to be more efficient than if tasks were performed sequentially by unspecialized individuals [188, 272].

In polymorphic species of ants, two (or more) physically different types of workers coexist. For example, in *Pheidole* species, minor workers are smaller and morphologically distinct from major workers. Minors and majors tend to perform differ-

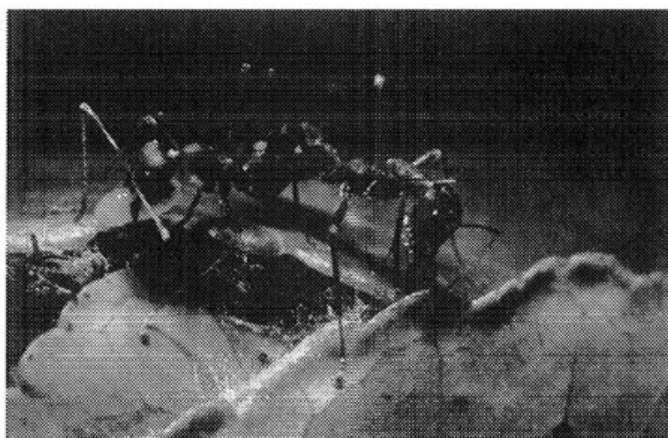


FIGURE 1.3 Two workers holding a larva in their mandibles.

ent tasks: whereas majors cut large prey with their large mandibles or defend the nest, minors feed the brood or clean the nest. Removal of minor workers stimulates major workers into performing tasks usually carried out by minors (see chapter 3, section 3.2) [330]. This replacement takes place within two hours of minor removal. More generally, it has been observed in many species of insects that removal of a class of workers is quickly compensated for by other workers: division of labor exhibits a high degree of plasticity.

Honey bees (*Apis mellifica*) build series of parallel combs by forming chains that induce a local increase in temperature. The wax combs can be more easily shaped thanks to this temperature increase [82]. With the combined forces of individuals in the chains, wax combs can be untwisted and be made parallel to one another. Each comb is organized in concentric rings of brood, pollen, and honey. Food sources are exploited according to their quality and distance from the hive (see section 1.2.2). At certain times, a honey bee colony divides: the queen and approximately half of the workers leave the hive in a swarm and first form a cluster on the branch of a nearby tree (Figure 1.4). Potential nesting sites are carefully explored by scouts. The selection of the nesting site can take up to several days, during which the swarm precisely regulates its temperature [166].

Nest construction in the wasp *Polybia occidentalis* involves three groups of workers, pulp foragers, water foragers, and builders. The size of each group is regulated according to colony needs through some flow of information among them (Figure 1.5) [189].

Tropical wasps (for example, *Parachartergus*, *Epipona*) build complex nests, comprised of a series of horizontal combs protected by an external envelope and connected to each other by a peripheral or central entrance hole (Figure 1.6) [187].

Some species of termites (*Macrotermes*) build even more complex nests (Figure 1.7), comprised of roughly cone-shaped outer walls that often have conspicuous

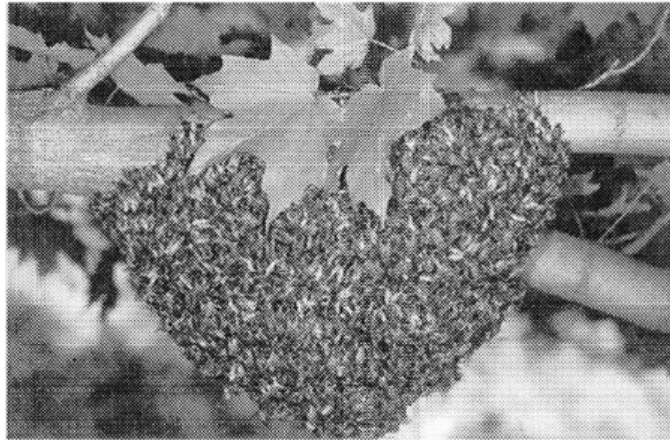


FIGURE 1.4 A swarm of honey bees *Apis mellifica*.

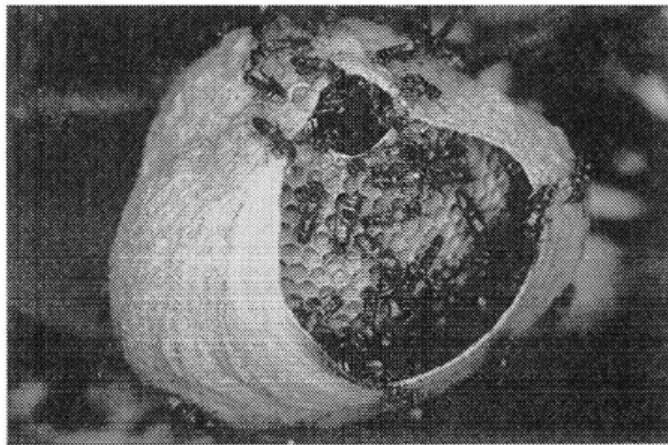


FIGURE 1.5 Nest building in the neotropical wasp *Polybia occidentalis*.

ribs containing ventilation ducts which run from the base of the mound toward its summit, brood chambers within the central “hive” area, which consists of thin horizontal lamellae supported by pillars, a base plate with spiral cooling vents, a royal chamber, which is a thick-walled protective bunker with a few minute holes in its walls through which workers can pass, fungus gardens, draped around the hive and consisting of special galleries or combs that lie between the inner hive and the outer walls, and, finally, peripheral galleries constructed both above and below ground which connect the mound to its foraging sites [227, 228].

And there are many more examples of the impressive capabilities of social insects.<sup>2</sup> If no one is in charge, how can one explain the complexity and sophistication

---

<sup>2</sup>Many more examples are discussed in greater detail in Camazine et al. [60].

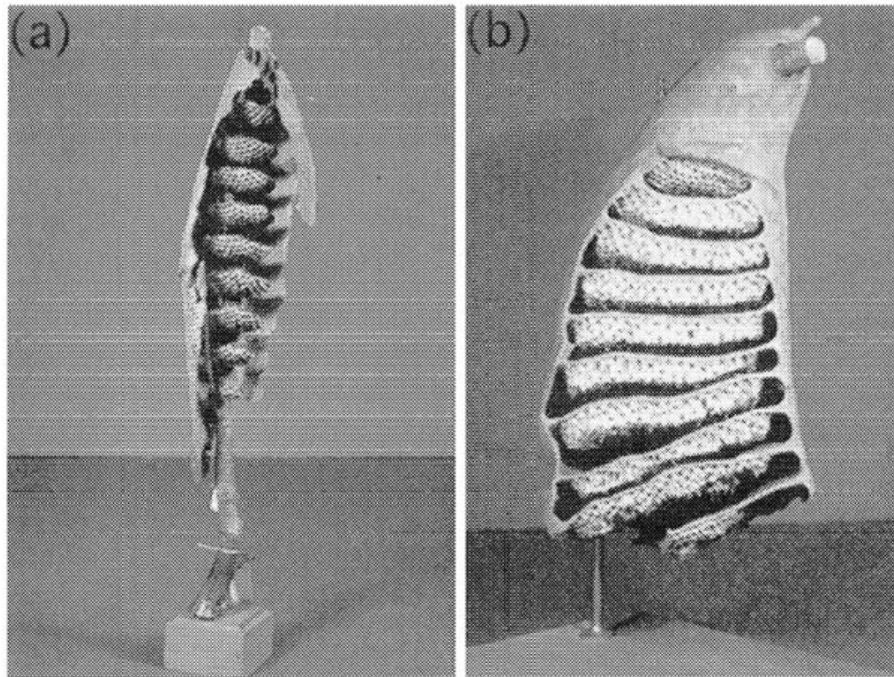


FIGURE 1.6 (a) Nest of a *Parachartergus* wasp species. (b) Nest of the wasp *Epipona tatua*.

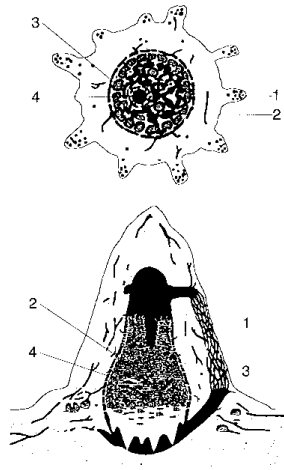


FIGURE 1.7 Cross section of a *Macrotermes* mound. (1) Walls containing ventilation ducts. (2) Brood chambers. (3) Base plate. (4) Royal chamber. After Lüscher [228]. Reprinted by permission © John L. Howard.

of their collective productions? An insect is a complex creature: it can process a lot of sensory inputs, modulate its behavior according to many stimuli, including interactions with nestmates, and make decisions on the basis of a large amount of information. Yet, the complexity of an individual insect is still not sufficient to explain the complexity of what social insect colonies can do. Perhaps the most difficult question is how to connect individual behavior with collective performance? In other words, how does cooperation arise?

Some of the mechanisms underlying cooperation are genetically determined: for instance, anatomical differences between individuals, such as the differences between minors and majors in polymorphic species of ants, can organize the division of labor. But many aspects of the collective activities of social insects are *self-organized*. Theories of self-organization (SO) [164, 248], originally developed in the context of physics and chemistry to describe the emergence of macroscopic patterns out of processes and interactions defined at the microscopic level, can be extended to social insects to show that complex collective behavior may emerge from interactions among individuals that exhibit simple behavior: in these cases, there is no need to invoke individual complexity to explain complex collective behavior. Recent research shows that SO is indeed a major component of a wide range of collective phenomena in social insects [86].

Models based on SO do not preclude individual complexity: they show that *at some level of description* it is possible to explain complex collective behavior by assuming that insects are relatively simple interacting entities. For example, flying from a food source back to the nest involves a set of complex sensorimotor mechanisms; but when one is interested in collective food source selection in honey bees (see section 1.2.2), the primitives that one may use in a model need not go into the detail of the implementation of flight, and flying back to the nest will be considered a simple behavior, although at the neurobiological level of description, flying back to the nest is certainly not simple. Nor does one need to use a model based on elementary particles to describe the aerodynamics of airplanes. Moreover, models based on SO are logically sound: they assume that it might be possible to explain something apparently complex in terms of simple interacting processes. If such an explanation is possible, then why does it have to be more complex? It is only when such an explanation fails that more complex assumptions will be put into the model.

The discovery that SO may be at work in social insects not only has consequences on the study of social insects, but also provides us with powerful tools to transfer knowledge about social insects to the field of *intelligent system design*. In effect, a social insect colony is undoubtedly a decentralized problem-solving system, comprised of many relatively simple interacting entities. The daily problems solved by a colony include finding food, building or extending a nest, efficiently dividing labor among individuals, efficiently feeding the brood, responding to external challenges, spreading alarm, etc. Many of these problems have counterparts in engineering and computer science. One of the most important features of social insects is that they can solve these problems in a very flexible and robust way: flexibility

allows adaptation to changing environments, while robustness endows the colony with the ability to function even though some individuals may fail to perform their tasks. Finally, social insects have limited cognitive abilities: it is, therefore, simple to design agents, including robotic agents, that mimic their behavior at some level of description.

In short, the modeling of social insects by means of SO can help design artificial distributed problem-solving devices that self-organize to solve problems—swarm-intelligent systems. It is, however, fair to say that very few applications of swarm intelligence have been developed. One of the main reasons for this relative lack of success resides in the fact that swarm-intelligent systems are hard to “program,” because the paths to problem solving are not predefined but emergent in these systems and result from interactions among individuals and between individuals and their environment as much as from the behaviors of the individuals themselves. Therefore, using a swarm-intelligent system to solve a problem requires a thorough knowledge not only of what individual behaviors must be implemented but also of what interactions are needed to produce such or such global behavior.

One possible idea is to develop a catalog of all the collective behaviors that can be generated with simple interacting agents: such a catalog would be extremely useful in establishing clear connections between artificial swarms and what they can achieve, but this could be a boring and endless undertaking. Another, somewhat more reasonable, path consists of studying how social insects collectively perform some specific tasks, modeling their behavior, and using the model as a basis upon which artificial variations can be developed, either by tuning the model parameters beyond the biologically relevant range or by adding nonbiological features to the model. The aim of this book is to present a few models of natural swarm intelligence and how they can be transformed into useful artificial swarm-intelligent devices.

This approach is similar to other approaches that imitate the way “nature” (that is, physical or biological systems) solves problems [37, 121]. Another possible, and apparently very promising, pathway is to use economic or financial metaphors to solve problems [182, 185, 196].

The expression “swarm intelligence” was first used by Beni, Hackwood, and Wang [13, 14, 15, 16, 161, 162] in the context of cellular robotic systems, where many simple agents occupy one- or two-dimensional environments to generate patterns and self-organize through nearest-neighbor interactions. Using the expression “swarm intelligence” to describe only this work seems unnecessarily restrictive: that is why we extend its definition to include any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies. And, strangely enough, this definition only marginally covers work on cellular robotic systems, which does not borrow a lot from social insect behavior.

From an historical perspective [2], the idea of using collections of simple agents or automata to solve problems of optimization and control on graphs, lattices, and networks was already present in the works of Butrimenko [53], Tsetlin [318], Stefanyuk [292], and Rabin [79, 265]. The latter introduced moving automata that

solve problems on graphs and lattices by interacting with the consequences of their previous actions [2]. Tsetlin [318] identified the important characteristics of biologically-inspired automata that make the swarm-based approach potentially powerful, randomness, decentralization, indirect interactions among agents, and self-organization. Butrimenko [53] applied these ideas to the control of telecommunications networks, and Stefanyuk [292] to cooperation between radio stations.

## 1.2 MODELING COLLECTIVE BEHAVIOR IN SOCIAL INSECTS

### 1.2.1 MODELING AND DESIGNING

The emphasis of this book is how to design adaptive, decentralized, flexible, and robust artificial systems, capable of solving problems, inspired by social insects. But one necessary first step toward this goal is understanding the mechanisms that generate collective behavior in insects. This is where modeling plays a role. Modeling is very different from designing an artificial system, because in modeling one tries to uncover what actually happens in the natural system—here an insect colony. Not only should a model reproduce some features of the natural system it is supposed to describe, but its formulation should also be consistent with what is known about the considered natural system: parameters cannot take arbitrary values, and the mechanisms and structures of the model must have some biological plausibility. Furthermore, the model should make testable predictions, and ideally all variables and parameters should be accessible to experiment.

Without going too deeply into epistemology, it is clear that an engineer with a problem to solve does not have to be concerned with biological plausibility: efficiency, flexibility, robustness, and cost are possible criteria that an engineer could use. Although natural selection may have picked those biological organizations that are most “efficient” (in performing all the tasks necessary to survival and reproduction), “flexible,” or “robust,” the constraints of evolution are not those of an engineer: if evolution can be seen as essentially a tinkering process, whereby new structures arise from, and are constrained by, older ones, an engineer can and must resort to whatever available techniques are appropriate. However, the remarkable success of social insects (they have been colonizing a large portion of the world for several million years) can serve as a starting point for new metaphors in engineering and computer science.

The same type of approach has been used to design artificial neural networks that solve problems, or in the development of genetic algorithms for optimization: if the brain and evolution, respectively, served as starting metaphors, most examples of neural networks and genetic algorithms in the context of engineering are strongly decoupled from their underlying metaphors. In these examples, some basic principles of brain function or of evolutionary processes are still present and are most important, but, again, ultimately a good problem-solving device does not have to be biologically relevant.

### 1.2.2 SELF-ORGANIZATION IN SOCIAL INSECTS

Self-organization is a set of dynamical mechanisms whereby structures appear at the global level of a system from interactions among its lower-level components. The rules specifying the interactions among the system's constituent units are executed on the basis of purely local information, without reference to the global pattern, which is an emergent property of the system rather than a property imposed upon the system by an external ordering influence. For example, the emerging structures in the case of foraging in ants include spatiotemporally organized networks of pheromone trails. Self-organization relies on four basic ingredients:

1. Positive feedback (amplification) often constitutes the basis of morphogenesis in the context of this book: they are simple behavioral “rules of thumb” that promote the creation of structures. Examples of positive feedback include recruitment and reinforcement. For instance, recruitment to a food source is a positive feedback that relies on trail laying and trail following in some ant species, or dances in bees.

When a bee finds a nectar source, she goes back to the hive and relinquishes her nectar to a hive bee. Then she can either start to dance to indicate to other bees the direction and the distance to the food source, or continue to forage at the food source without recruiting nestmates, or she can abandon her food source and become an uncommitted follower herself. If the colony is offered two identical food sources at the same distance from the nest, the bees exploit the two sources symmetrically. However, if one source is better than the other, the bees are able to exploit the better source, or to switch to this better source even if discovered later. Let us consider the following experiment (Figure 1.8). Two food sources are presented to the colony at 8:00 a.m. at the same distance from the hive: source A is characterized by a sugar concentration of 1.00 mol/l and source B by a concentration of 2.5 mol/l. Between 8:00 and noon, source A has been visited 12 times and source B 91 times. At noon, the sources are modified: source A is now characterized by a sugar concentration of 2.5 mol/l and source B by a concentration of 0.75 mol/l. Between noon and 4:00 p.m., source A has been visited 121 times and source B only 10 times.

It has been shown experimentally that a bee has a relatively high probability of dancing for a good food source and abandoning a poor food source. These simple behavioral rules allow the colony to select the better quality source. Seeley et al. [287] and Camazine and Sneyd [59] have confirmed with a simple mathematical model based on these observations that foragers can home in on the best food source through a positive feedback created by differential rates of dancing and abandonment based upon nectar source quality. Figure 1.9 shows a schematic representation of foraging activity; decision points (C1: become a follower?) and (C2: become a dancer?) are indicated by black diamonds. Figure 1.10(a) shows the number of different individuals that visited each feeder during the previous half hour in the experiments. Figure 1.10(b) shows the forager group size (here,

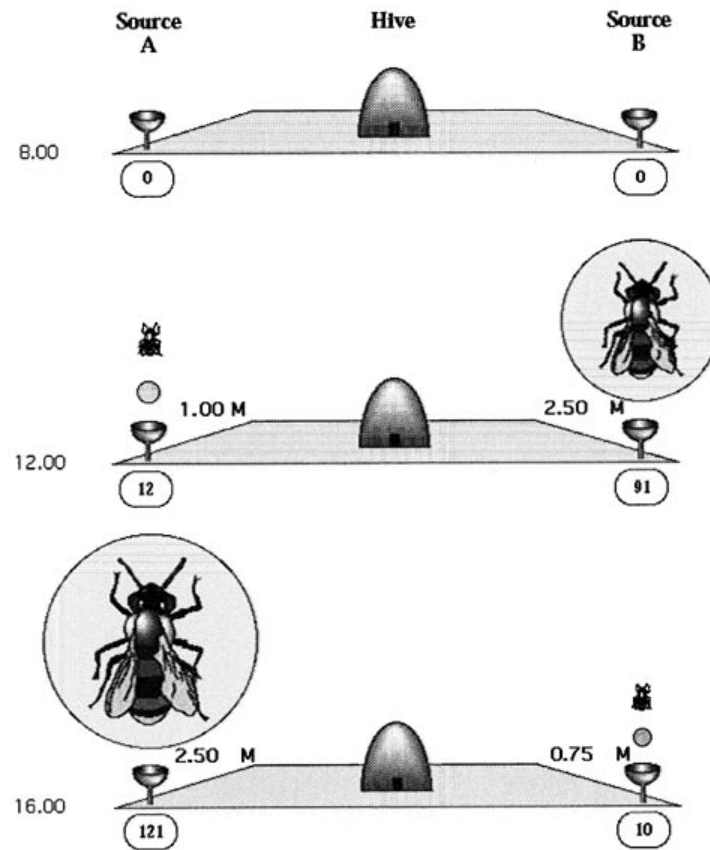


FIGURE 1.8 Schematic representation of the experimental setup.

the sum of the bees dancing for the feeder, the bees at the feeder, and the bees unloading nectar from the feeder) for each feeder obtained from simulations of the simple model depicted in Figure 1.9.

2. Negative feedback counterbalances positive feedback and helps to stabilize the collective pattern: it may take the form of saturation, exhaustion, or competition. In the example of foraging, negative feedback stems from the limited number of available foragers, satiation, food source exhaustion, crowding at the food source, or competition between food sources.
3. SO relies on the amplification of fluctuations (random walks, errors, random task-switching, and so on). Not only do structures emerge despite randomness, but randomness is often crucial, since it enables the discovery of new solutions, and fluctuations can act as seeds from which structures nucleate and grow. For example, foragers may get lost in an ant colony, because they follow trails with some level of error; although such a phenomenon may seem inefficient, lost foragers can find new, unexploited food sources, and recruit nestmates to these food sources.

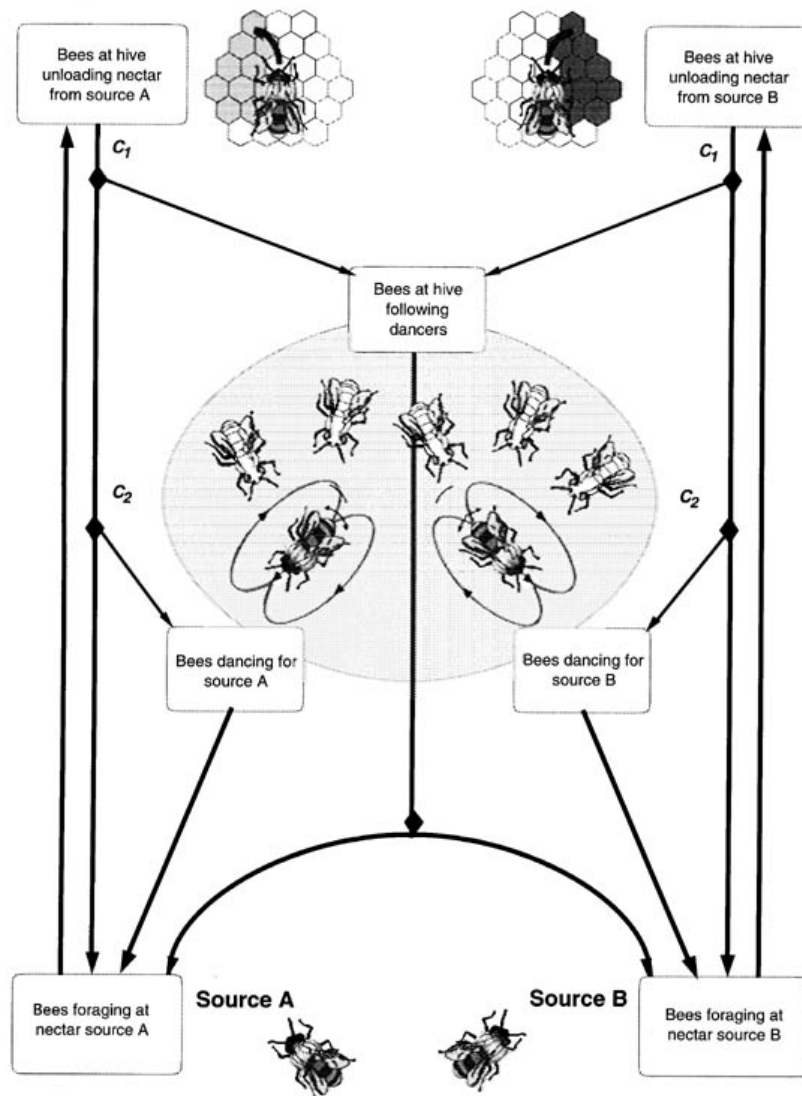


FIGURE 1.9 Schematic representation of foraging activity.

4. All cases of SO rely on multiple interactions. A single individual can generate a self-organized structure such as a stable trail provided pheromonal<sup>3</sup> lifetime is sufficient, because trail-following events can then interact with trail-laying actions. However, SO generally requires a minimal density of mutually tolerant individuals. Moreover, individuals should be able to make use of the results of their own activities as well as of others' activities (although they may perceive

<sup>3</sup>A pheromone is a chemical used by animals to communicate. In ants, a pheromone trail is a trail marked with pheromone. Trail-laying trail-following behavior is widespread in ants (see chapter 2)

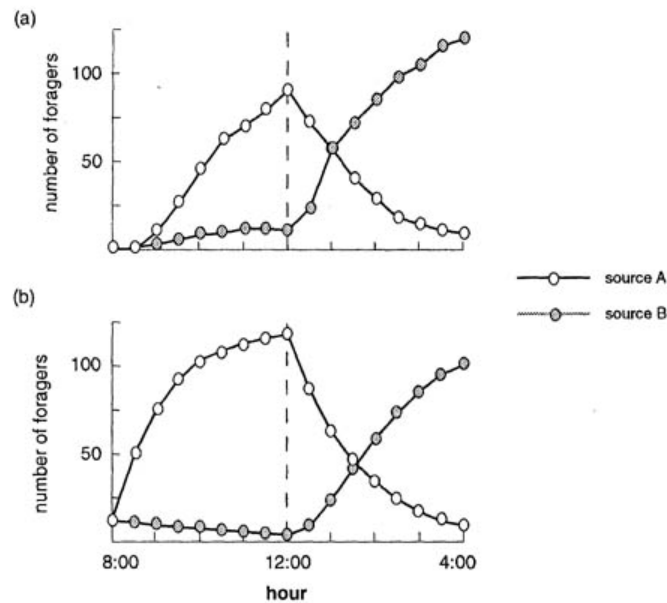


FIGURE 1.10 (a) Experimental number of different individuals that visited each feeder during the previous half hour. (b) Forager group size for each feeder obtained from model.

the difference): for instance, trail networks can self-organize and be used collectively if individuals use others' pheromone. This does not exclude the existence of individual chemical signatures or individual memory which can efficiently complement or sometimes replace responses to collective marks.

When a given phenomenon is self-organized, it can usually be characterized by a few key properties:

1. The creation of spatiotemporal structures in an initially homogeneous medium. Such structures include nest architectures, foraging trails, or social organization. For example, a characteristic well-organized pattern develops on the combs of honeybee colonies. This pattern consists of three concentric regions (a central brood area, a surrounding rim of pollen, and a large peripheral region of honey) (Figure 1.11). It results to a large extent from a self-organized process based on local information [58]. The model described in Camazine [58] relies on the following assumptions, suggested by experimental observations:
  - (a) The queen moves more or less randomly over the combs and lays most eggs in the neighborhood of cells already occupied by brood. Eggs remain in place for 21 days.
  - (b) Honey and pollen are deposited in randomly selected available cells.
  - (c) Four times more honey is brought back to the hive than pollen.

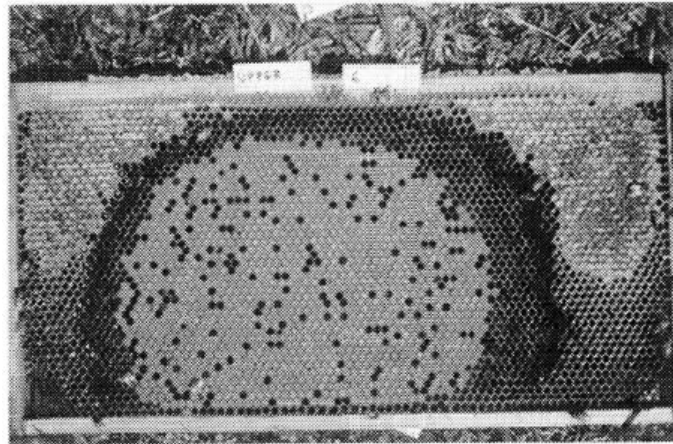


FIGURE 1.11 Comb of a manmade hive: the three concentric regions of cells can be clearly seen.

(d) Typical removal input ratios for honey and pollen are 0.6 and 0.95, respectively.

(e) Removal of honey and pollen is proportional to the number of surrounding cells containing brood.

Simulations of a cellular automaton based on these assumptions were performed [58]. Figure 1.12 shows six successive steps in the formation of the concentric regions of brood (black dot on a white background), pollen (dark grey), and honey (light grey). Assumptions 1a and 1e ensure the growth of a central compact brood area if the first eggs are laid approximately at the center of the comb. Honey and pollen are initially randomly mixed [assumption 1b], but assumptions 1c and 1d imply that pollen cells are more likely to be emptied and refilled with honey, so that pollen located in the periphery is removed and replaced by honey. The only cells available for pollen are those surrounding the brood area, because they have a high turnover rate. The adaptive function of this pattern is discussed by Camazine [58].

2. The possible coexistence of several stable states (multistability). Because structures emerge by amplification of random deviations, any such deviation can be amplified, and the system converges to one among several possible stable states, depending on initial conditions. For example, when two identical food sources, A and B, are presented at the same distance from the nest to an ant colony that resorts to mass recruitment,<sup>4</sup> one of them is eventually massively exploited while the other is neglected: both sources have the same chance of being exploited, but only one of them is, and the colony could choose either one. There are therefore two possible attractors in this example: massive exploitation of A, or

<sup>4</sup>Mass recruitment in ants is based solely on trail-laying trail-following.

massive exploitation of B. Which attractor the colony will converge to depends on random initial events.

3. The existence of bifurcations when some parameters are varied. The behavior of a self-organized system changes dramatically at bifurcations. For example, the termite *Macrotermes* uses soil pellets impregnated with pheromone to build pillars. Two successive phases take place [157]. First, the noncoordinated phase is characterized by a random deposition of pellets. This phase lasts until one of the deposits reaches a critical size. Then, the coordination phase starts if the group of builders is sufficiently large: pillars or strips emerge. The existence of an initial deposit of soil pellets stimulates workers to accumulate more material through a positive feedback mechanism, since the accumulation of material reinforces the attractivity of deposits through the diffusing pheromone emitted by the pellets [48]. This autocatalytic, “snowball effect” leads to the coordinated phase. If the number of builders is too small, the pheromone disappears between two successive passages by the workers, and the amplification mechanism cannot work; only the noncoordinated phase is observed. Therefore, there is no need to invoke a change of behavior by the participants in the transition from the noncoordinated to the coordinated phase: it is merely the result of an increase in group size. A more detailed account of pillar construction in *Macrotermes* can be found in chapter 5 (section 5.2).

### 1.2.3 STIGMERGY

Self-organization in social insects often requires interactions among insects: such interactions can be direct or indirect. Direct interactions are the “obvious” interactions: antennation, trophallaxis (food or liquid exchange), mandibular contact, visual contact, chemical contact (the odor of nearby nestmates), etc. Indirect interactions are more subtle: two individuals interact indirectly when one of them modifies the environment and the other responds to the new environment at a later time. Such an interaction is an example of stigmergy. In addition to, or in combination with, self-organization, stigmergy is the other most important theoretical concept of this book. Grassé [157, 158] introduced stigmergy (from the Greek *stigma*: sting, and *ergon*: work) to explain task coordination and regulation in the context of nest reconstruction in termites of the genus *Macrotermes*. Grassé showed that the coordination and regulation of building activities do not depend on the workers themselves but are mainly achieved by the nest structure: a stimulating configuration triggers the response of a termite worker, transforming the configuration into another configuration that may trigger in turn another (possibly different) action performed by the same termite or any other worker in the colony. Nest reconstruction consists of first building strips and pillars with soil pellets and stercoral mortar; arches are then thrown between the pillars and finally the inter-pillar space is filled to make walls. Figure 1.13 sketches how Grassé’s [157, 158] notion of stigmergy can be applied to pillar construction.

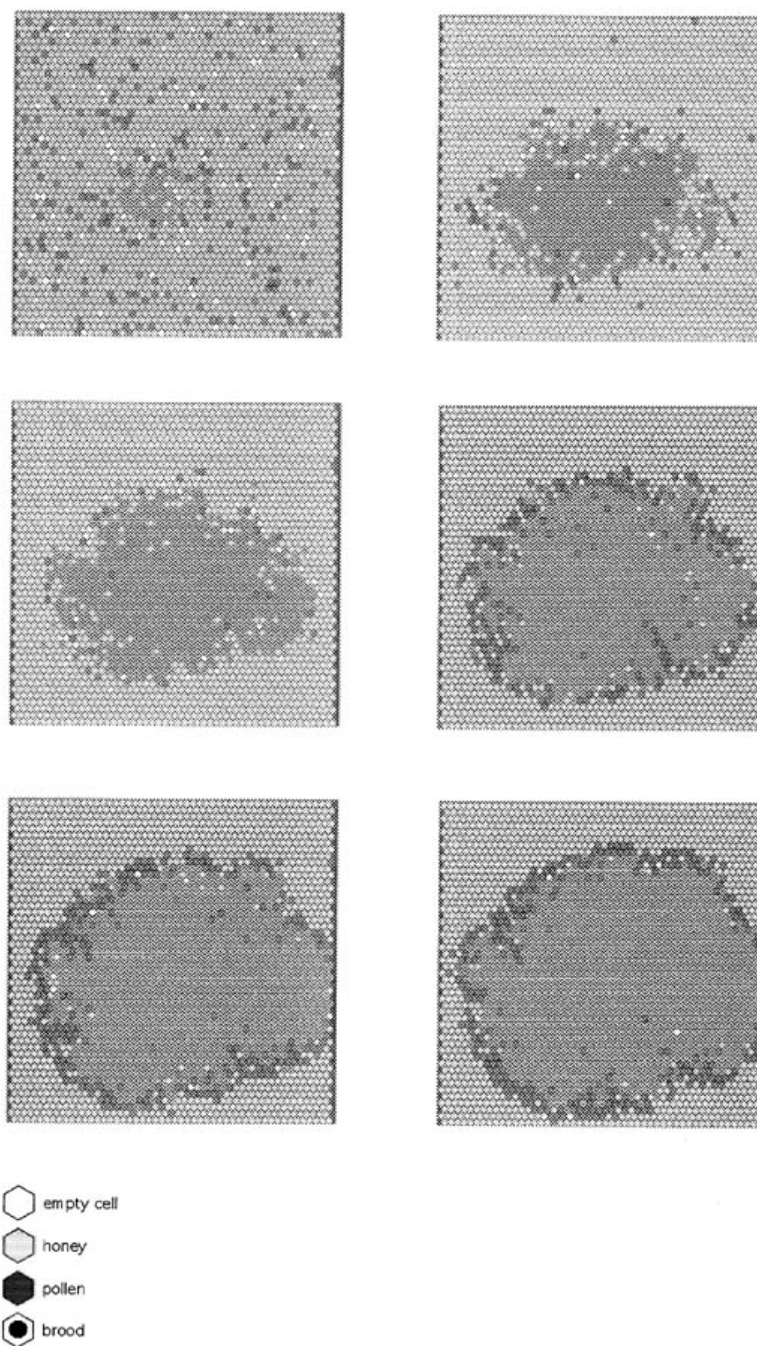


FIGURE 1.12 Six successive steps in the formation of the concentric regions of brood (black dot in a white background), pollen (dark grey), and honey (light grey). Empty cells are represented in white.

Stigmergy is easily overlooked, as it does not explain the detailed mechanisms by which individuals coordinate their activities. However, it does provide a general mechanism that relates individual and colony-level behaviors: individual behavior modifies the environment, which in turn modifies the behavior of other individuals.

The case of pillar construction in termites shows how stigmergy can be used to coordinate the termites' building activities by means of self-organization. Another illustration of how stigmergy and self-organization can be combined is recruitment in ants, described in more detail in chapter 2: self-organized trail laying by individual ants is a way of modifying the environment to communicate with nestmates that follow such trails. In chapter 3, we will see that task performance by some workers decreases the need for more task performance: for instance, nest cleaning by some workers reduces the need for nest cleaning. Therefore, nestmates communicate to other nestmates by modifying the environment (cleaning the nest), and nestmates respond to the modified environment (by not engaging in nest cleaning): that is stigmergy. In chapters 4 and 5, we describe how ants form piles of items such as dead bodies, larvae, or grains of sand. There again, stigmergy is at work: ants deposit items at initially random locations. When other ants perceive deposited items, they are stimulated to deposit items next to them. In chapter 6, we describe a model of nest building in wasps, in which wasp-like agents are stimulated to deposit bricks when they encounter specific configurations of bricks: depositing a brick modifies the environment and hence the stimulatory field of other agents. Finally, in chapter 7, we describe how insects can coordinate their actions to collectively transport prey. When an ant changes position or alignment, it modifies the distribution of forces on the item. Repositioning and realignment cause other transporting ants to change their own position or alignment. The case for stigmergy is a little more ambiguous there, because the item could be considered a medium of "direct" communication between ants. Yet, the same mechanism as in the other examples is at work: ants change the perceived environment of other ants. In every example, the environment serves as a medium of communication.

What all these examples have in common is that they show how stigmergy can easily be made operational. That is a promising first step to design groups of artificial agents which solve problems: replacing coordination through direct communications by indirect interactions is appealing if one wishes to design simple agents and reduce communication among agents. Another feature shared by several of the examples is incremental construction: for instance, termites make use of what other termites have constructed to contribute their own piece. In the context of optimization, incremental improvement is widely used: a new solution is constructed from previous solutions. Finally, stigmergy is often associated with flexibility: when the environment changes because of an external perturbation, the insects respond *appropriately* to that perturbation, as if it were a modification of the environment caused by the colony's activities. In other words, the colony can *collectively* respond to the perturbation with individuals exhibiting the same behavior. When it comes to artificial agents, this type of flexibility is priceless: it means that the agents can

respond to a perturbation without being reprogrammed to deal with that particular perturbation.

### 1.3 MODELING AS AN INTERFACE

As was discussed in section 1.2.1, understanding nature and designing useful systems are two very different endeavors. Understanding nature requires observing, carrying out experiments, and making models that are constrained by observations and experiments. Designing, on the other hand, requires making models that are only limited by one's imagination and available technology. But, in both cases, models are of utmost importance, although they are sometimes implicitly, rather than explicitly, and verbally, rather than mathematically, formulated. Discussing the essence of modeling is definitely beyond the scope of this book. However, it is worth giving a brief outline of what it means to model a phenomenon because models are central to the work presented here. Indeed, models of natural phenomena in social insects set the stage for artificial systems based on them. For the purpose of this book, a model is a simplified picture of reality: a usually small number of observable quantities, thought to be relevant, are identified and used as variables; a model is a way of connecting these variables. Usually, a model has parameters, that, ultimately, should be measurable. Hidden variables, that is, additional assumptions which are not based on observable quantities, are often necessary to build a connection among variables that is consistent with observed behavior. Varying the values of the parameters usually modifies the output or behavior of the model. When the model's output is consistent with the natural system's behavior, parameter values should be compared with the values of their natural counterparts whenever possible. Sources of hidden variables are looked for in the natural system. A good model has several qualities: parsimony, coherence, refutability, etc.

Models channel imagination in the sense that they usually contain the ingredients necessary to explain classes of phenomena: it might be worth exploring models which are known to (plausibly) explain phenomena of interest rather than start from scratch. Exploring a model beyond the constraints of reality amounts to using a tool beyond its initial purpose. Knowing that social insects have been particularly successful at solving problems which can be abstracted away and formulated in, say, algorithmic language, makes models of problem solving in social insects particularly attractive: models that help explain how social insects solve problems serve as a starting point because they can be explored beyond their initial boundaries. For instance, optimization algorithms based on models of foraging often resort to pheromone decay over time scales that are too short to be biologically plausible: in this case, models have been explored beyond what is biologically relevant to generate an interesting new class of algorithms. The underlying principles of these algorithms are strongly inspired by the functioning of social insect colonies, but some of the parameter values do not lie within empirical boundaries. In the context of this book, modeling serves, therefore, as an interface between understanding

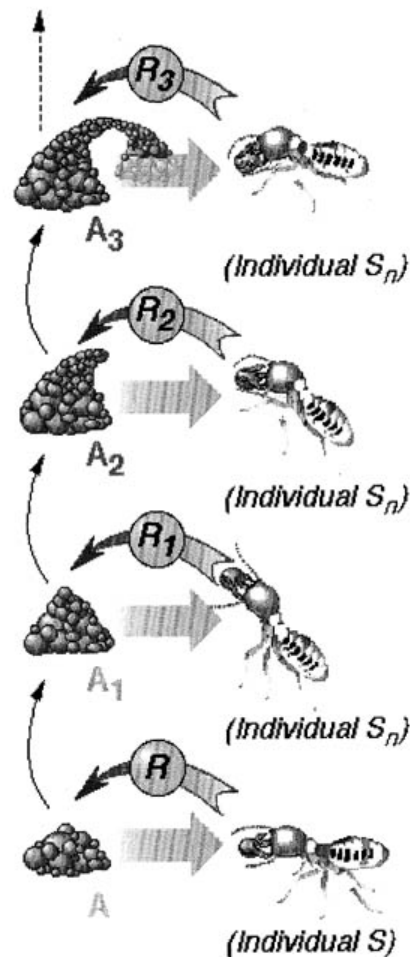


FIGURE 1.13 Assume that the architecture reaches state  $A$ , which triggers response  $R$  from worker  $S$ .  $A$  is modified by the action of  $S$  (for example,  $S$  may drop a soil pellet), and transformed into a new stimulating configuration  $A_1$ , that may in turn trigger a new response  $R_1$  from  $S$  or any other worker  $S_n$  and so forth. The successive responses  $R_1, R_2, R_n$  may be produced by any worker carrying a soil pellet. Each worker creates new stimuli in response to existing stimulating configurations. These new stimuli then act on the same termite or any other worker in the colony. Such a process, where the only relevant interactions taking place among the agents are indirect, through the environment which is modified by the other agents, is also called *sematectonic* communication [329]. After Grassé [158]. Reprinted by permission © Masson.

nature and designing artificial systems: one starts from the observed phenomenon, tries to make a biologically motivated model of it, and then explores the model without constraints.

## 1.4 FROM ALGORITHMS TO ROBOTICS

Swarm-based robotics is growing so rapidly that it is becoming very difficult to keep track of what is going on and to keep up with novelty. A recent issue of the journal *Autonomous Robots* was dedicated to “colonies” of robots. Other major journals in the field of robotics have already published, or are in the process of publishing special issues on this topic. Our aim is not to provide the reader with an extensive survey of the field,<sup>5</sup> but rather to present a few selected examples in which researchers have clearly made use of the social insect metaphor, or the swarm-based approach, through one or all of the following: the design of distributed control mechanisms, the type of coordination mechanisms implemented, the tasks that the robots perform. This will be sufficient to have an idea of the advantages and pitfalls of reactive distributed robotics, which can also be called, in the present context, swarm-based robotics.

Why is swarm-based robotics, that we would define loosely as reactive collective robotics, an interesting alternative to classical approaches to robotics? Some of the reasons can be found in the characteristic properties of problem solving by social insects, which is flexible, robust, decentralized, and self-organized. Some tasks may be inherently too complex or impossible for a single robot to perform [62] (see, for example, chapter 7, where pushing a box requires the “coordinated” efforts of at least two individuals). Increased speed can result from using several robots, but there is not necessarily any cooperation: the underlying mechanism that allows the robots to work together is a minimum interference principle. Designing, building, and using several simple robots may be easier because they use, for example, a simpler sensorimotor apparatus, cheaper because they are simple, more flexible without the need to reprogram the robots, and more reliable and fault-tolerant because one or several robots may fail without affecting task completion—although completion time may be affected by such a perturbation—than having a powerful complex robot. Furthermore, theories of self-organization [31] teach us that, sometimes, collective behavior results in patterns which are qualitatively different from those that could be obtained with a single agent or robot. Randomness or fluctuations in individual behavior, far from being harmful, may in fact greatly enhance the system’s ability to explore new behaviors and find new “solutions.” In addition, self-organization and decentralization, together with the idea that interactions among agents need not be direct but can rather take place through the environment, point to the possibility of significantly reducing communication between robots: explicit robot-to-robot communication rapidly becomes a big issue when the number of robots increases; this

---

<sup>5</sup>Such reviews—thorough at the time they were released and already somewhat obsolete—can be found, for example, in Cao et al. [61, 62].

issue can be, to a large extent, eliminated by suppressing such communication! Also, central control is usually not well suited to dealing with a large number of agents (this is also true for a telecommunications network: see section 2.7 in chapter 2), not only because of the need for robot-to-controller-and-back communications, but also because failure of the controller implies failure of the whole system.

Of course, using a swarm of robots has some drawbacks. For example, stagnation is one (see chapter 7): because of the lack of a global knowledge, a group of robots may find itself in a deadlock, where it cannot make any progress. Another problem is to determine how these so-called “simple” robots should be programmed to perform user-designed tasks. The pathways to solutions are usually not predefined but emergent, and solving a problem amounts to finding a trajectory for the system and its environment so that the states of both the system and the environment constitute the solution to the problem: although appealing, this formulation does not lend itself to easy programming. Until now, we implicitly assumed that all robots were identical units: the situation becomes more complicated when the robots can perform different tasks, respond to different stimuli, or respond differently to the same stimuli, and so forth; if the body of theory that roboticists can use for homogeneous groups of robots is limited, there is virtually no theoretical guideline for the design of heterogeneous swarms.

The current success of collective robotics is the result of several factors:

1. The relative failure of the Artificial Intelligence (AI) program, which “classical” robotics relied upon, has forced many computer scientists and roboticists to reconsider their fundamental paradigms; this paradigm shift has led to the advent of connectionism, and to the view that sensorimotor “intelligence” is as important as reasoning and other higher-level components of cognition. Swarm-based robotics relies on the anti-classical-AI idea that a group of robots may be able to perform tasks without explicit representations of the environment and of the other robots; finally, planning is replaced by reactivity.
2. The remarkable progress of hardware during the last decade has allowed many researchers to experiment with real robots, which have not only become more efficient and capable of performing many different tasks, but also cheap(er).
3. The field of Artificial Life, where the concept of emergent behavior is emphasized as essential to the understanding of fundamental properties of the living, has done much to propagate ideas about collective behavior in biological systems, particularly social insects; theories that had remained unknown to roboticists have eventually reached them; of course, the rise of the internet-based “global information society” enhanced this tendency.
4. Finally, collective robotics has become fashionable by positive feedback. The field is booming, but not that many works appear to be original, at least conceptually, which is the only level the authors of this book can judge.

Interesting directions for the future include ways of enhancing indirect communications among robots. For example, Devezza et al. [95] introduced odor sensing

for robot guidance (which, of course, is reminiscent of trail following in ants), Russell [278] used a short-lived heat trail for the same purpose. Prescott and Ibbotson [264] used a particularly original medium of communication to reproduce the motion of prehistoric worms: bathroom tissue. A dispenser on the back of the robot releases a stream of paper when the robot is moving; two light detectors on each of the side arms measure reflected light from the floor and control the thigmotaxis (toward track) and phobotaxis (away from track) behaviors. Although their work is not about collective robotics, it sets the stage for studying and implementing indirect communication in groups of robots.

Finally, what are the areas of potential application for swarm-based multirobot systems? “In aerospace technology, it is envisioned that teams of flying robots may effect satellite repair, and aircraft engine maintenance could be performed by thousands of robots built into the engine eliminating the need for costly disassembly for routine preventive maintenance. Environmental robots are to be used in pipe inspection and pest eradication. While industrial applications include waste disposal and micro cleaners. Ship maintenance and ocean cleaning could be performed by hundreds of underwater robots designed to remove debris from hulls and ocean floors. . . . Some researchers envision microsurgical robots that could be injected into the body by the hundreds designed to perform specific manipulation tasks without the need for conventional surgical techniques,” writes Kube [208] (p. 17). Most of these applications require miniaturization. Very small robots, micro- and nano-robots, which will by construction have severely limited sensing and computation, may need to “operate in very large groups or swarms to affect the macroworld [237].” Approaches directly inspired or derived from swarm intelligence may be the only way to control and manage such groups of small robots. As the reader will perhaps be disappointed by the “simplicity” of the tasks performed by state-of-the-art swarm-based robotic systems such as those presented in chapters 4, 6, and 7, let us remind him that it is in the perspective of miniaturization that swarm-based robotics becomes meaningful.

In view of these great many potential applications, it seems urgent to work at the fundamental level of what algorithms should be put into these robots: understanding the nature of coordination in groups of simple agents is a first step toward implementing useful multirobot systems. The present book, although it deals marginally with collective robotics, provides a wealth of ideas that, we hope, will be useful in that perspective.

## 1.5 READING GUIDE

It may be useful at this point to issue a series of warnings to the reader. This book is not intended to provide the reader with recipes for solving problems. It is full of wild speculations and statements that may (or may not) turn out to be just plain wrong. This book is not describing a new kind of algorithmic theory. Nor does it provide