13

Harnessing the swarm: technological applications of collective intelligence

Suzanne Sadedin and Edgar A. Duéñez-Guzmán

One of the most influential concepts in artificial intelligence is the notion of the swarm. That is, intelligent adaptive behaviour can arise in large groups of interacting agents, even when the individual agents have limited local information and use simple rules. Self-organisation provides a basic structure in such agent societies, while natural selection can drive the evolution of increasingly efficient and coordinated interactions through improved communication, information processing, and agent specialisation. Such collective intelligences have evolved in diverse biological contexts, ranging from foraging and home-building colonies of ants, termites and bees, to the coordinated movements of vertebrate flocks and schools, to the exquisitely tuned dynamical responses of immune and neural systems. Here, we discuss how these biological models contribute to emerging technologies in fields such as optimisation, robotics, image processing, self-repairing systems and automatic structure design.

The main issues

Many modern engineering designs have been based on natural adaptations, a procedure termed biomimicry (Benyus, 2002). Among the more ambitious of these designs are those that incorporate the selective process itself. By evolving solutions to problems, researchers aim to capture the robust and adaptive properties of organisms. The growing complexity of information technology demands machines and algorithms with the ability to respond flexibly and intelligently to new situations without supervision, a feature common in living systems but virtually unheard of in normal engineering.

Pragmatic Evolution: Applications of Evolutionary Theory, ed. Aldo Poiani. Published by Cambridge University Press.

David Green describes in Chapter 12 how natural selection can be used to solve difficult problems via evolutionary algorithms. Here, we will consider how evolutionary theory can contribute to technology more broadly through swarm intelligence (Beni, 2005; Bonabeau and Theraulaz, 2008; Camazine *et al.*, 2003; Krause and Ruxton, 2002).

At the most general level, swarm intelligence arises through the interaction of two key evolutionary processes, natural selection and self-organisation. The process of natural selection underlies countless specialised adaptations that solve complex problems involved in survival and reproduction of living organisms. Self-organisation occurs when the behaviour of units within a system contributes to behaviour at the level of the whole system, without any global controller regulating these units. Although selection drives evolution, self-organisation provides the raw material of phenotypes upon which selection acts to generate adaptation, creating continuous feedback between the two processes (see reviews by Kauffman, 1993; Halley and Winkler, 2008). Selection improves the genes that regulate these complex and dynamic networks, but it does so only through their expression in self-organised physical and chemical structures, the phenotypes among which selection will choose.

Collective intelligence in nature

Collective intelligence is seen in diverse natural systems, ranging from unicellular organisms to human societies. The classical inspiration for collective intelligence is the colonial behaviour of the eusocial insects (Figure 13.1D,E): ants, termites, bees and wasps whose workers are sterile. For example, a colony of leaf-cutter ants digs, maintains and defends a well-organised nest with discrete chambers for queens, larvae, fungus farms and garbage dumps, locates and collects leaves to feed its underground fungus farms, continuously rears new generations of workers in several castes, and eventually reproduces (Hölldobler and Wilson, 1990). Yet there is no central controller of the ants' nest, nor does any single ant know the impact of its actions on the nest. Instead, each ant makes simple choices based on a small number of rules and the information in its immediate environment. The rules used by the ants are shaped by natural selection through their consequences for the colony, which emerge through self-organisation. Bees, wasps and termites form similar nests, ranging in size from a few individuals to many thousands. Termite colonies can include millions of individuals and are elaborately structured to control temperature, moisture, oxygen and carbon dioxide balance (Emerson, 1938; Korb, 2003). Both ants and termites use caste systems, where individuals take different, specialised developmental paths in response to nutritional cues, allowing highly effective division of labour within the colony; individual ants may even specialise in vocations within their cast (Julian and Cahan, 1999).

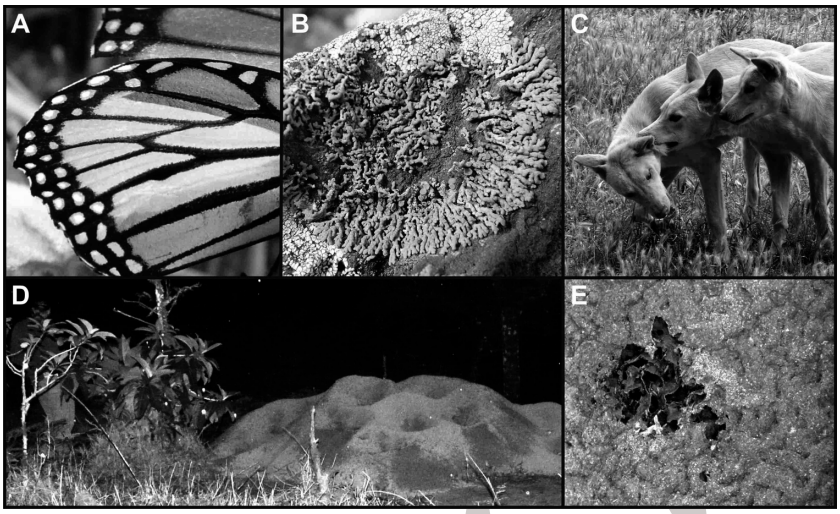

Figure 13.1 Intelligence emerges from the interaction of self-organisation and selection. (A) A monarch butterfly's wing evolved through selection to repel predators. Its detailed pattern develops through self-organisation via local communication among cells. (B) A lichen effectively explores space using simple branching rules. (C) Dingoes, like many vertebrates, cooperate in social groups, but these groups require complex negotiation of individual interests that undermine their collective intelligence. (D) In contrast, ants cooperate to build large, efficient colonies. (E) A termite nest, showing the complex internal structure which develops through the simple decisions of individual termites.

Perhaps the most striking example of collective intelligence is the integrated action of cells within multicellular organisms (Figure 13.1A) which construct and maintain complex organs for chemical and information transportation and processing using only local information. Most notably, the collective actions of large numbers of interacting neurons provide the definitive standard for intelligence. Likewise, the vertebrate immune system continually adapts to identify and eliminate new infections. Such cellular interactions are not normally considered as collective intelligences. However, recent reviews have noted that the underlying mechanisms for cooperative and adaptive behaviour seem to be the same, whether for cells within individuals or individuals within a superorganism (Couzin, 2009). For this reason, we will also discuss these systems as examples of collective intelligence.

Limited collective intelligence also occurs in other contexts. Many organisms that are capable of living as single cells, such as bacteria, slime moulds and simple fungi, will cooperate to form structured colonies that effectively explore spatial environments or facilitate the flow of nutrients (Figure 13.1B). Multiple species may be involved in these arrangements (as seen in the

complex ecological communities of stromatolites; Papineau *et al.*, 2005). Cellular slime moulds breed as free-living amoeba in moist environments, but during dry periods they aggregate to form a single cooperative entity (Bonner, 1971). Examples of collective intelligence in vertebrates (Figure 13.1C) include the coordinated movements of fish schools, hunts by orca pods, collaborative defence by rook flocks and the complex societies of *Homo sapiens*. Such groups are generally far from acting as a superorganism. Rather, they represent alliances based on a combination of kinship and common interest, often fraught with individual conflicts that can cause whole groups to fragment. Nonetheless, flocks of birds, schools of fish and herds of mammals often make rapid, intelligent decisions as a group that outperform individual choices (Couzin *et al.*, 2005). Computational research has revealed simple rules that often underlie these decisions (Reynolds, 1987; Heppner and Grenander, 1990). For example, a well-coordinated flock can emerge when individuals follow three rules: (1) steer in the average direction of neighbours; (2) steer toward the average position of neighbours; (3) avoid crowding at close range.

Common trends have been noted among these varying forms of collective intelligence (Garnier *et al.*, 2007; Couzin, 2009; Krause *et al.*, 2009). First, advanced collective intelligence usually involves large numbers of simple units. Yet there is also a role for functional differentiation: brain modules, ant and termite castes, and different immune cells all exhibit division of labour. However, such division of labour seems to be confined to highly cooperative groups. For cells in multicellular animals, and for eusocial insects, there is generally no alternative but to cooperate; they cannot reproduce independently. This commitment to common goals allows a high level of differentiation and specialisation, increasing efficiency at the level of the organism or superorganism. In contrast, single-celled organisms and most multicellular organisms are capable of independent reproduction, generating evolutionary conflict among individuals. Cooperation among such individuals requires careful negotiation, so although they may be highly intelligent individually, they exhibit less intelligence as collectives. Human societies present an exception to this generalisation: through trade, we harness our competitive needs and individual intelligence to build functional societies more complex than any individual can comprehend.

Another important mechanism of collective intelligence is stigmergy (Beckers *et al.*, 1994). Rather than storing information internally and communicating directly, simple units will often modify their environment so that it stimulates an appropriate response in other units. For example, models suggest that the complex galleries and walled chambers of termite nests emerge through environmental feedback from gas exchange and pheromone trails, without any change in individual decision-making (Bonabeau, 1998).

Lastly, collective intelligence itself often involves natural selection. The behaviour of individual units is usually highly stochastic, causing varying behaviour that allows the system as a whole to explore the variety of possible solutions to a particular problem in parallel. The system then chooses between candidate solutions by communication among its units. The non-linear dynamics of interactions among units allow meaningful local information to become amplified by positive feedback loops and thus spread rapidly through the network, but dampen unstructured noise (Couzin, 2009). In this way, rewarding areas of solution space are explored in more detail, while unrewarding areas are gradually abandoned. For example, within the brain, neurons initially connect arbitrarily, but connections that induce rewards grow thicker and recruit additional connections, while those that go unrewarded are starved and fade away. Similarly, the immune system produces diverse, widely varying cells, but selectively propagates the few that succeed in identifying infections. Ants in search of food wander in a random walk, but once they find it, they emit chemical signals which recruit nestmates to form trails; consequently, ant trails to better and nearer food sources grow thicker and wider, while trails to worse sources die off (Deneubourg *et al.*, 1990a). Even schooling fish swim in the direction chosen by larger numbers of more enthusiastic fish, effectively avoiding danger and locating food sources that might be beyond the range of any individual, while automatically correcting for errors (Couzin *et al.*, 2005).

Theory of collective intelligence

The natural examples described above suggest that sophisticated intelligence emerges from the interactions of large numbers of units that have been structured by selection to act in coordination for the attainment of a shared goal. Moreover, in each example described, there are components of both small random changes and selection among candidate solutions, providing an analogue of natural selection in each form of problem-solving. The mechanics of this process, however, remain elusive. To understand how swarms can solve problems intelligently, we need to look more deeply at what it means to be intelligent. Note that this section involves some advanced mathematical ideas; readers who prefer to skip to the 'Looking forward' section can do so safely without losing the thread of the discussion.

What problems are hard, and why?

The usefulness of computers lies in their ability to provide us with solutions to problems we pose them. In general terms, *computable* means that when presented with a task (such as finding the median in a collection of values), we can produce a program that solves the problem (computes and outputs

the median). There are, however, some problems that cannot be solved by any computer program, and are called *non-computable* or *undecidable* (for a detailed treatment of algorithmic complexity, see Sipser, 2005; Fortnow and Homer, 2002). An example is the *halting problem*: no computer program can decide whether any given program will run forever or eventually halt.

While all computable problems have a solution in the strictest sense, in reality it is important to focus on *tractability*. Intuitively, a problem is *intractable* if a computer would take too much time to solve it. In the theory of computation, the most-used measure of tractability is running time as a function of the size of the input problem. In other words, how long would it take in the worst-case scenario if the problem to solve were twice as big? Would it be twice as slow? Ten times? More? This concept is called asymptotic running time and is usually represented by the big-O notation (Knuth, 1997; Sipser, 2005). We use $O(f(n))$ to denote that the running time is at most proportional to $f(n)$. For instance, the best known algorithm for finding the median of a set of numbers has a running time of $O(n)$ (Cormen *et al.*, 2003). That is, if an input of n (fixed size) integers is given, the algorithm can find the median in time proportional to n. This running time includes the overhead incurred in reading the input in the first place (which is $O(n)$ running time).

Most real-world problems have a complexity that is somewhere in between $O(n)$ and undecidability. A problem is considered tractable if there is a k such that a program that solves it has running time of $O(n^k)$. These problems are referred to as P, for they can be solved in *polynomial* time by a computer. While it is clear that there are algorithms with a running time that is not polynomial, there are still many problems for which there is no known polynomial time solution. Note that there is a subtle difference between problems and algorithms here. An algorithm has a certain running time that can usually be established by analysing its code. A problem can have very many (or infinite) algorithms that solve it, but when considering its tractability we focus on the *best* possible solution.

The most important class of problems for which no known polynomial time solution exists are called *NP-complete* (NP stands for non-deterministic polynomial time). If these were proven to not be in P (the famous $P \neq NP$ question), then even the best algorithm would take (essentially) exponential time to solve them. This is extremely important because exponentials grow prohibitively big very quickly. Table 13.1 illustrates how, even though exponential time starts small, it grows to surpass a polynomial very quickly. In fact, *any* exponential will necessarily surpass *any* polynomial given a big enough n. For instance, in our example it would only take an input of $n = 60$ for the running time to be well above the existence time of the universe, while n^5 would require $n \approx 3500$. This disparate growth rate is the reason to consider non-polynomial time intractable.

Table 13.1 Exponential versus polynomial time. Assume the time is given in seconds. Note how the exponential grows quickly and surpasses a polynomial.

n	n^5	Time	2^n	Time
10	10^5	~1 day	1024	~17 minutes
20	3.2×10^6	~1 month	1.04×10^6	~12 days
30	2.43×10^7	~10 months	1.07×10^9	~34 years
40	1.02×10^8	~3 years	1.09×10^{12}	~34 000 years

Finally, while the $P\neq NP$ inequality has not been proven yet, virtually every mathematician and computer scientist believes this to be the case. Moreover, NP-complete problems arise very commonly in real-world problems. Perhaps the best known of these is the *Travelling Salesman Problem* (TSP). Imagine you are a salesman and want to travel to some cities to promote a product. Being an efficient salesman, you do not want to visit the same city twice, and want to minimise your travel distance. This simple problem is NP-complete and if $P\neq NP$, it can only be solved in at least exponential time, making it intractable. See Chapter 12 for a more complete discussion of the TSP.

How can swarms help?

In the previous section, we established that there are some important and hard problems, for which finding the best solution is unreasonable in terms of running time. But not all is lost. Sometimes in practical terms it suffices to find a *good* solution, not necessarily the *best* solution. This approach is called *heuristic*. Take, for instance, the TSP described above. If an exact algorithm takes 2 years to find a route of 100 km, we may well prefer an heuristic algorithm that takes 5 minutes to find a route of 102 km.

With this in mind, researchers have turned their attention to heuristics as a compromise between optimality of solutions and time to obtain them. These approximate algorithms have been inspired by a multitude of natural and artificial processes. Evolution by natural selection provides one way to search for solutions to problems (for example, the evolutionary algorithms discussed in Green's chapter). However, many species have also evolved the ability to find reasonable solutions to intractable problems in real time, using a range of heuristic methods. Moreover, because they have evolved through selection in nature, the heuristics used by living organisms are well-suited to practical problem-solving in the natural world; they tend to be robust, adaptable and quick. In traditional problem-solving, if an aspect of a solution becomes unfeasible or the problem changes, a new solution often

needs to be obtained from scratch. However, many nature-based stochastic algorithms are resilient to such changes, and the algorithm can usually continue execution without major impact on the solution.

However, there is yet another advantage of swarms over other heuristics. That is, the model itself need not be simulated within a single computer. With availability of cheap and small devices of growing computational power, swarms may become the computational model of choice. Traditional computer science views computers as *deterministic Turing machines* to study solutions to problems; tractability is, in effect, a problem's solvability in polynomial time with a deterministic Turing machine (P). However, there is a conceptually alternative computational model, the *non-deterministic Turing machine*, which considers a computer that, when presented with a choice, simulates all possible decisions at the same time. Tractability, then, effectively becomes solvability in polynomial time with a non-deterministic Turing machine (NP). The $P \neq NP$ problem can be thought of as asking whether a deterministic Turing machine can simulate polynomial steps of a non-deterministic Turing machine in polynomial time. Now consider a swarm of computers as your computational model. Start with one individual, and each time a computation requires a branching, let an individual recruit others to compute that branch of the non-deterministic algorithm. Such branching effectively shifts the computing paradigm from a serial to a massively parallel and effectively non-deterministic approach. The question of $P \neq NP$ then becomes irrelevant.

Traditional multiprocessor computing has been done through clusters. These consist of several interconnected computers that a programmer uses in parallel to solve a problem. The main limitation of clusters is that the programmer needs to specify how each computer is used. Grid-computing removes this limitation using a swarm-inspired approach; a problem to be solved is automatically divided and sent to many computers, each one specialised in solving a particular subproblem. A solution is then integrated from its parts and sent back to the requesting source. More recently, several applications for handheld devices have emerged that run in the background and process only small amounts of information individually, but solve a big problem collectively. Similarly, self-organisation and communication can optimise efficiency: for example, Kassabalidis *et al.* (2001) suggested that nodes in a network, like the Internet, could use simple rules to locally decide to which neighbours to connect, thus self-organising to decrease transmission bottlenecks, increase availability of resources and improve scalability. By the end of the 1980s, a shift of focus in robotics had also occurred, away from classical (completely determined, exact solutions) and toward biologically inspired solutions (simple rules, emergent behaviour). These approaches have proved very successful and form the basis for modern robotics research (Brooks, 1990).

Limitations of swarms

Although heuristics have proven to be very effective and simple approaches to problem-solving, considerably more effort has been devoted to study exact solutions theoretically. Consequently, we have yet to develop clear performance criteria to choose among heuristics. Studying algorithmic complexity, running time and alternative models of computation seem natural choices to evaluate exact solutions, but these concepts do not translate directly to heuristics. Because heuristics are faster than exact solutions at providing near-optimal solutions, running time and algorithmic complexity are usually ignored when comparing heuristics. Instead, researchers focus on the accuracy of the solution obtained; however, this may not always be relevant.

The theory of *No Free Lunch* (NFL) is perhaps the most successful attempt at answering the question of whether a heuristic is, in fact, better than another one. By considering heuristics as black-box algorithms (i.e. algorithms that assume nothing about the problem they are solving), one can abstract all the properties of the algorithm and focus on the path the algorithm traverses through the search space of possible solutions. While there are several NFL theorems that are applicable in different scenarios, the original formulation states that given no information about which problem to solve, any search path is as good at exploring the solution space as any other (Wolpert and Macready, 1997; Schumacher *et al.*, 2001). Viewed in this way, the NFL seems hardly surprising: given no information, no action can be expected to be better than any other. The usual conclusion is that, on average, all algorithms perform the same. While originally NFL theorems were concerned with all search (Wolpert and Macready, 1995) and optimisation algorithms and average peformance over all possible functions, multiple extensions to machine learning, subsets of algorithms and functions and other cases have been proposed, improving our understanding of the theoretical expectations and behavior of heuristics (Whitley and Watson, 2005; Whitley and Rowe, 2008; Rowe *et al.*, 2009).

A concrete example of such limits is given by Krause *et al.* (2009). They presented human groups with two problems: estimating the number of marbles in a jar, and calculating a simple probability. While the mean estimate for the first problem was remarkably accurate, the mean estimate for the second problem was highly misleading due to systematic biases. Thus the swarm is useful only when its units and their interactions are shaped to solve relevant problems, and the major challenge of swarm research is to fine-tune simple interaction rules and developmental paths to obtain intelligent behaviour in a relevant context. Since evolution has had millions of years to find systems that are effective in nature, identifying relevant natural adaptive problems and their evolved solutions remains a crucial shortcut. However, even in biological systems, the relevant features are often hard to discern.

Simulated evolutionary and adaptive approaches to finding rules that allow swarms to complete specific tasks are increasingly popular (Dorigo *et al.*, 2004; Trianni *et al.*, 2004) to circumvent this problem. We noted previously that some form of natural selection is often incorporated at a proximate level in natural collective intelligences. Natural selection, although no more robust than other heuristics in true black-box searching, can be advantageous in situations where better solutions are clustered along some dimension because it encourages searching near relatively productive areas.

Looking forward

Many applications of swarm intelligence are currently in development. The research programs inspired by predictions from evolutionary theory have exposed a wealth of complex adaptations that underlie numerous ongoing technological innovations in robotics and artificial intelligence. Here we discuss such specific applications of swarms, and their links to evolutionary theory. However, to truly exploit the potential of swarms, we need to understand how intelligent behaviour emerges from the interaction of units under selection. This area is still in its infancy as a research field, but several exciting developments point to possible future technologies with potential to transform many aspects of human life. While there are dangers in excessive speculation, history suggests it is equally perilous to ignore the possible impacts of such transformative technologies (see Green *et al.*, 2010). Therefore, we will also survey some of these more speculative technological applications.

Artificial neural networks (ANNs)

The field of ANNs was founded as early as the 1940s, when Pitts and Mcculloch (1947) developed the first model of artificial neurons. Later, Rosenblatt (1958) proposed the perceptron as a neuron with many variable-weight inputs and one output, that fires when its inputs exceed a threshold. Networks of perceptrons (Figure 13.2) were analysed in detail by Minsky (1967) and Minsky and Papert (1987), who described the classes of problem solvable with them and their mathematical properties. Later, Hornik *et al.* (1989) showed mathematically that ANNs are universal approximators; that is, they are capable, with the right topology, to compute any function with arbitrary precision. There are two key requirements for an ANN to solve a problem successfully (for a detailed review, see Rojas, 1996). First, the *topology* of the network defines which problems the ANN can solve and which it cannot. Second, the network must *learn*: that is, the weights of the neuron connections (synapses) need to be modified in such a way that they can robustly process the input.

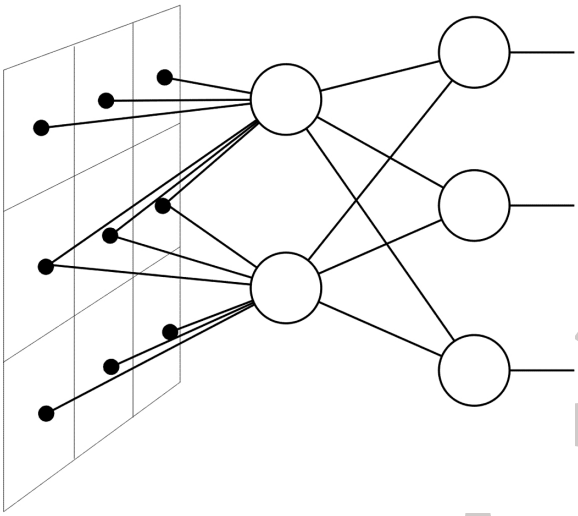

Figure 13.2 Schematic view of a network of perceptrons. In this example, we have a 3×3 grid that acts as an input layer to the network. The network itself has two clear layers (no cycles). The last layer is called the output layer, and in this case has three perceptrons or neurons.

Evolutionary theory and swarm intelligence have been used both to design topology and train networks (Yao, 1993; Cliff *et al.*, 1992; Juang, 2004). Long-standing applications of ANNs have been in pattern recognition (Ripley, 1996) and classification (Carpenter *et al.*, 1991), as well as forecasting (Zhang *et al.*, 1998), and data projection and visualisation (Su and Chang, 2001). Today, ANNs are increasingly applied in robotics, for example to learn handling objects and other desirable behaviours (Ito *et al.*, 2006). Only recently has the computational power been available to construct an ANN the size of the human neocortex (the part of the brain responsible of language, sensory processing and consciousness) (Johansson and Lansner, 2007). By combining these technologies with better algorithms to find the right topology for problem-solving, including evolutionary methods, we are perhaps approaching the development of the first artificial generalised intelligence.

Ant colony optimisation

In the 1970s and 1980s it was established that most species of foraging ants lay down a chemical compound called a pheromone when travelling from and to food sources. Deneubourg *et al.* (1990a) showed how a foraging

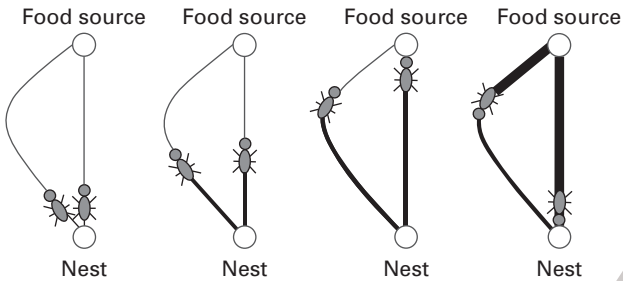

Figure 13.3 Foraging ants are presented with two possible paths to a food source. While walking, ants leave behind a chemical marker that can be detected by other ants. Ants following the shortest path return faster to the nest, thus strengthening that path even more. The figure illustrates the amount of pheromone laid on the two paths (the thicker the line, the more pheromone it contains) as time progresses.

path could be established by ants following simple rules of exploration based on concentrations of pheromone laid by other ants. An ant presented with several paths is more likely to follow the one that has a higher concentration of pheromone. As a consequence of this simple rule, and the physical properties of the environment, a colony of ants is capable of finding the shortest route to a food source. For example, if there are two paths of different length to a food source from the nest, an ant taking the shortest path is likely to return faster and thus increase the amount of pheromone laid on that trail (see Figure 13.3). Consequently, more ants will follow that path than the other one. Also, pheromone evaporates over time so that if later a new, shorter path is discovered, or if the known path is blocked, then the current path can be abandoned in favour of other solutions.

Termites, slime mould, and molluscs have all been shown to use similar chemical signals. The generality of the approach highlights a tug-of-war between two concepts inherent to all heuristics: exploration vs. exploitation (see Chapter 12). We can think of the process of laying the pheromone as strengthening exploitation, while the imperfect selection of a path by an ant and the evaporation of pheromone laid down can be seen as maintaining exploration.

Originally introduced by Dorigo *et al.* (1991), the initial application of artificial ant colonies was in solving combinatorial optimisation problems such as the TSP (Colorni *et al.*, 1991). The geometric intuition and shortest-path properties of foraging ants translated directly to these problems. Later, ant colony optimisation was applied in a wide variety of domains from scheduling problems (Merkle *et al.*, 2002; Blum and Dorigo, 2004) to data mining (Parpinelli *et al.*, 2002). Some recent applications include topics

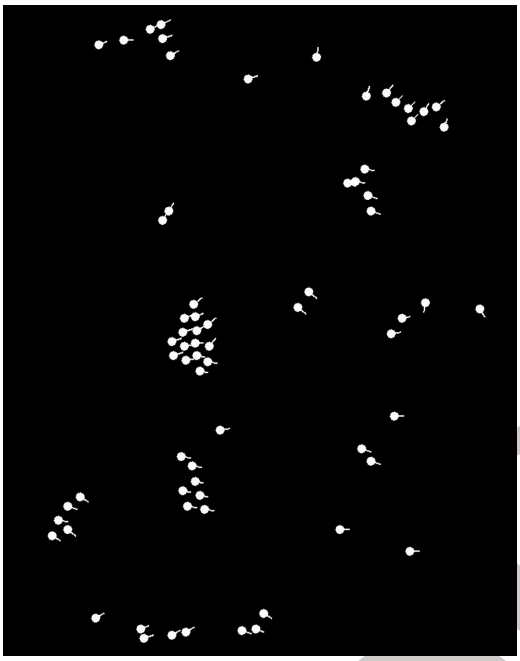

Figure 13.4 A simple simulated flock, where individuals adjust their trajectory and velocity based on that of others (http://vlab.infotech.monash.edu.au/).

as diverse as learning Bayesian network equivalence classes (Daly and Shen, 2009), antenna design (Galehdar *et al.*, 2009), fuzzy-system design (Juang *et al.*, 2009), Bayesian network structure learning (Pinto *et al.*, 2009), routing protocols in mobile networks (Wu and Song, 2008), cancer gene discovery (Xiong and Wang, 2009), signal transmission reliability (in terms of jamming resistance; Zaka *et al.*, 2008) and optimal clustering (Handl and Meyer, 2007).

Particle swarms

During the late 1980s, the flocking, herding and schooling behaviours of birds, mammals and fish began to receive active attention from computer scientists and theoretical biologists (Reynolds, 1987; Heppner and Grenander, 1990). Inspired by the self-organisation of complex coordination seen in ant colonies, simple rules were hypothesised that could explain the complex behaviours of flocks (Figure 13.4). Further investigation revealed that such self-organised flocks of simple agents could also at times make better decisions than single individuals. This led to the suggestion by Kennedy and Eberhart (1995) that simulations of flock-like

collectives could be used for problem-solving, a technique that became known as *Particle Swarm Optimisation* (PSO).

In PSO, much like in other evolutionary algorithms (see Chapter 14), a population of individuals (or *particles* in PSO jargon) is simulated, with each individual representing a potential solution to a problem. However, unlike other evolutionary algorithms, the dynamics of individuals in PSO are modelled from social interactions and flocking behaviour instead of being shaped by simulated natural selection and adaptation (see review by Poli *et al.*, 2007). An individual in PSO is considered to have a position (coding for a solution to the problem), a velocity and (partial) memory of previous positions. Using local information on the state of neighbours and their own state and memory, individuals update their velocity following simple rules.

The rules for updating a particle's velocity are intuitively given by adjusting its velocity in the direction of its best-known previous position, and the position of the particle with the best position in its neighbourhood. These changes are, however, weighted randomly and independently, and some noise is also introduced to the velocity to add diversity. In addition, there are several ways to define the neighbours of a particle, for instance by having a fully connected neighbourhood (everyone affects everyone else) or other static neighbourhood structures. Alternatively, it is possible to allow the neighbourhood to change dynamically, based for instance on Euclidean distance among particles, random subsets, or quality of solutions.

Some current applications of PSO include job scheduling on computational grids and data mining (Abraham *et al.*, 2006), fuzzy system optimisation (Juang *et al.*, 2009), design reliability optimisation (Muñoz Zavala *et al.*, 2005c), registering 3D-to-3D biomedical images (Wachowiak *et al.*, 2004), solving structural design problems (Perez and Behdinan, 2007), controlling reactive power and voltage (Yoshida *et al.*, 2000), and analysing human tremor (Eberhart and Hu, 1999). Combining evolutionary algorithms (see Chapter 12) with particle swarm ideas by treating the particles as individuals under selection has also proven a fruitful approach (Angeline, 1998; Muñoz Zavala *et al.*, 2005a, 2005b). Particle swarms have even been used to evolve the weights and topology of neural networks (Kennedy and Eberhart, 1995; Eberhart *et al.*, 2001; Eberhart and Shi, 2001).

Swarmbots

Swarms of robots, or 'swarmbots', may be used to solve many problems in the real world (Beni, 2005). Deneubourg *et al.* (1990b) originally proposed using swarms of ant-like robots to solve a sorting and classification problem spatially (Figure 13.5). He showed that simple individuals with

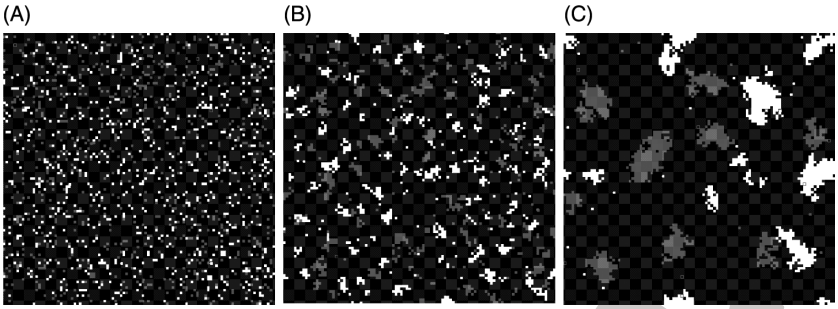

Figure 13.5 An algorithm for sorting objects, inspired by sorting behavior seen in ants. We start with a well-mixed set of blocks (A). Ants pick up blocks at random, and drop them when they encounter a block of the same colour. Consequently, piles of similar colour grow by positive feedback (B), eventually sorting the blocks into a few large, homogeneous piles (C) (http://vlab.infotech.monash.edu.au/).

local information and simple rules could solve complex problems at a global scale. This continues to be an active area of research (Handl and Meyer, 2007), but several other applications have been proposed and are being actively explored. There have also been attempts at formalising the behaviours of swarmbots by physical (Spears *et al.*, 2004) or logical properties (Winfield *et al.*, 2005), as well as assessing their robustness (Winfield and Nembrini, 2006).

Developing useful swarmbots poses formidable challenges at several levels. From a practical perspective, the construction of potentially thousands of physical robots needs to be economically feasible. Each robot needs to be complex enough to be able to carry out its part of the task, but simple enough that it is cheap and simple to build and re-program. The stress on low cost is twofold. On the one hand, there is the explicit need to build many of the robots, but on the other hand, there is an implicit assumption that some of these robots might be destroyed in the process of solving a problem. Swarmbots also present conceptual difficulties. Robots need to be able to interact and communicate with each other either directly or indirectly. The rules each robot is following need to be designed carefully to ensure the successful completion of the task. Evolutionary theory might offer the solution to this problem by artificially evolving behavioural and communication rules for the swarms (Dorigo *et al.*, 2004; Trianni *et al.*, 2004).

Current research has successfully used swarmbots to carry out simple tasks, and many new approaches are being developed, improved and extended. Some examples of these include exploration of unknown (and potentially dangerous or inhospitable) territory (Correll and Martinoli, 2006; Hsiang *et al.*, 2003); building arbitrary structures (Werfel *et al.*, 2005) from

pre-fabricated modules; self-repair of structures created with swarmbots as their building blocks (Rubenstein and Shen, 2009; Shen *et al.*, 2004); transportation of objects much bigger than one individual robot (Kube and Bonabeau, 2000); and maintaining formation while travelling through landscapes (Turgut *et al.*, 2008), and more (Şahin, 2005). One ambitious project involves a patient swallowing the pieces of a self-assembling robot to perform a surgical operation from within (Harada *et al.*, 2010).

Perhaps the most intriguing feature of swarmbots is their potential for self-assembly (Arbuckle and Requicha, 2004). Already, simple robots have been developed that use the materials they find in their environment to construct copies of themselves. Any self-replicating system may be subject to error, potentially initiating a new line of evolution independent of (and perhaps superseding) DNA-based life. Such engineered organisms present both dangers and irresistible opportunities. By creating custom life forms, we may ultimately harness self-replicating cellular machinery to generate virtually unlimited quantities of substances, limited only by the availability of raw resources. Photosynthetic nanobots, growing in sunlit vats and supplied with just an appropriate mud, could refine metals, absorb pollution, supply nutritious food to humans and livestock, and produce all the drugs and medications humanity can use. Many of these tools are already applied on small scales; the limitation is that current biological engineering relies on small modifications to existing organisms, rather than the thorough redesign of the organism needed for efficient and independent performance.

Releasing engineered organisms into the environment would enable even grander schemes. Free-roaming nanobots could coordinate to attack diseases as they found them, curing people before they knew they were sick. Whole ecosystems could be regulated, enabling farming without the use of pesticides, invasive organisms could be suppressed, and pollution could be removed wherever it was generated. Exploring and terraforming other planets for human habitation might be possible by seeding a planet with swarmbots or bacteria engineered to survive in the environment, produce a breathable atmosphere, moderate climates and remove toxins from the soil (Freitas, 1983; Imre Friedmann and Ocampo-Friedmann, 1995). Wandering robots could collaborate to construct and repair homes, roads, farms, factories, power plants and other infrastructure, while others would deliver them with materials mined and refined by still others. In this way, whole cities could be constructed by swarmbots.

However, the dangers of this approach are equally cogent. Entities that reproduce and evolve are not easily controlled. Without rigorous control over selection, it may evolve toward a form that is less efficient, useless or dangerous. The most disturbing aspect of such visions is the 'grey goo' problem: a nanobot which evolves a generalised ability to metabolise

organic matter, potentially converting all living organisms into copies of itself (Drexler, 1986). Later research suggested that grey goo is unlikely, and a number of techniques to avoid it have been proposed (Giles, 2004). Unfortunately, the history of human management of such novel risks is notoriously poor (see discussion by Green *et al.*, 2010). So even if it is theoretically possible to avoid the risks of uncontrolled evolution, human error may prevent us doing so.

Human swarms

Harnessing the swarm may also be a powerful mechanism for the construction of intelligent social systems and infrastructure. Globalisation implies that technologies used locally often spread throughout the world rapidly; consequently, interactions between devices form a broad network that presents both risks and opportunities. On the one hand, interaction can allow communication and cooperation. On the other hand, frequent interactions can give rise to unexpected consequences that change behaviour of entire systems dramatically at critical points, as local effects ripple through to create global consequences. Such phase changes, when they arise in systems subject to natural selection, may greatly alter long-term evolutionary dynamics (Green *et al.*, 2006). While considerable progress has been made in understanding network structure and function, the practical application of these network processes is still in its infancy as a field (Green *et al.*, 2010). By studying the individual behaviour that contributes to organisation in societies of cells and animals, we may find lessons in adaptively shaping the interactions of devices, algorithms and even human societies.

Many commonplace devices already communicate and interact to improve user experiences. This enhanced interactivity has led to unprecedented social phenomena such as flash mobs – public collective performance art organised by internet and mobile phone communication (Nicholson, 2005). The internet also enables more efficient use of human collective intelligence. Search engines collect outlinks from web pages because the resultant statistics direct searchers to information far more effectively than any expert rating (Langville *et al.*, 2008). Reputation systems, representing collective intelligence of a community about individuals, are used to enforce social standards on websites (Resnick *et al.*, 2000). Wikipedia, written and edited in small pieces by thousands of unpaid, self-selected individuals, now approaches the Encyclopedia Britannica in accuracy and far exceeds it in scope (Giles, 2005). Recently, a collective calling itself D.H.J. Polymath, who collaborated only via weblog comments, found in six weeks a new and simpler proof of the density Hales–Jewett theorem (Polymath, 2010). Researchers in digital business ecosystems are working on computational

systems to optimise the automatic evolution of trade relationships among organisations using design principles from evolution and ecology (Briscoe *et al.*, 2007).

Conclusions

We have discussed many insights from nature which, seen through the lens of evolutionary theory, are contributing to ongoing technological development. It is important to note that this is not a one-way interaction. The ongoing development of evolutionary theory, and the insights it provides into nature, are increasingly fed by technological innovations which open up new approaches to modelling and data handling. Many of the swarm approaches to artificial intelligence described above are now being applied to better understand biological phenomena such as the structure and function of genomes and the development of individuals. In a sense, this represents a 'snowballing' of knowledge: the more we learn about biology, the better we can develop technology, while the better our technology, the more we can understand biology.

The future holds many challenges for swarm research. In the near term, we urgently require theoretical developments to better understand how cooperation, individuation and communication can be reliably harnessed for problem-solving. Advances in evolutionary theory, including game theory, sociobiology and self-organisation, will form a crucial part of this work. Continuing development in robotics and materials science suggests that we will soon live in a world of largely autonomous, self-maintaining and self-healing devices. More broadly, a sophisticated control of swarm and robotic technologies offers unprecedented potential for low-cost chemical and material production, product manufacturing and construction and maintenance of every form of human infrastructure.

This potential, however, must be carefully exploited to manage evolutionary risks in an acceptable way. Globalisation generates increasing interaction and interdependence of nations, industries and societies. In his influential book *Collapse*, Diamond (2006) argued that many past civilisations have collapsed due to specific practices which ultimately destroyed the environment on which they depended. These collapses, as devastating as they were for the people involved, were nonetheless local events confined to a specific area and culture. Today, new practices and technology can easily spread globally long before their ultimate effects are known. As a species, this means we can no longer afford the errors of the past; if one group makes a crucial error, it is likely that the consequences will percolate to all of us. Advanced evolutionary theory will be crucial to managing these risks.

Acknowledgements

We thank David G. Green and Aldo Poiani for helpful discussion and comments. This work was supported by the Australian Research Council.

REFERENCES

Abraham, A., Guo, H. and Liu, H. 2006. *Swarm Intelligence: Foundations, Perspectives And Applications*. Studies in Computational Intelligence, vol. **26**. Berlin/Heidelberg: Springer Berlin/Heidelberg, pp. 3–25.

Angeline, P. 1998. Evolutionary optimization versus particle swarm optimization: philosophy and performance differences. *In*: Porto, V., Saravanan, N., Waagen, D. and Eiben, A. (eds), *Lecture Notes in Computer Science: Evolutionary Programming VII*. Springer, Berlin, **1447**:601–10.

Arbuckle, D. and Requicha, A. 2004. Active self-assembly. *In*: *IEEE International Conference on Robotics and Automation*, vol. **1**. Citeseer, Princeton, NJ, pp. 896–901.

Beckers, R., Holland, O.E. and Deneubourg, J.L. 1994. From local actions to global tasks: stigmergy and collective robotics. *In*: *Artificial Life IV*, vol. **181**. Citeseer, Princeton, NJ, p. 189.

Beni, G. 2005. From swarm intelligence to swarm robotics. *In*: Sahinm E. and Spears, W. (eds), *Lecture Notes in Computer Science: Swarm Robotics*. Springer, Berlin, pp. 1–9.

Benyus, J.M. 2002. *Biomimicry: Innovation Inspired by Nature*. Harper Perennial, New York, NY.

Blum, C. and Dorigo, M. 2004. The hyper-cube framework for ant colony optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part B* **34**:1161–72.

Bonabeau, E. 1998. A model for the emergence of pillars, walls and royal chambers in termite nests. *Philosophical Transactions of the Royal Society B: Biological Sciences* **353**(1375):1561.

Bonabeau, E. and Theraulaz, G. 2008. Swarm Smarts. *Scientific American* **18**:40–7.

Bonner, J.T. 1971. Aggregation and differentiation in the cellular slime molds. *Annual Reviews in Microbiology* **25**:75–92.

Briscoe, G., Sadedin, S. and Paperin, G. 2007. Biology of applied digital ecosystems. *In*: *Digital Ecosystems and Technologies Conference*. IEEE, New Jersey, pp. 458–463.

Brooks, R.A. 1990. Elephants don't play chess. *Robotics and Autonomous Systems* **6**:3–15.

Camazine, S., Deneubourg, J.L., Franks, N.R., *et al.* 2003. *Self-organization in Biological Systems*. Princeton University Press, Princeton, NJ.

Carpenter, G., Grossberg, S. and Reynolds, J. 1991. ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks* **4**:565–88.

Cliff, D., Harvey, I. and Husbands, P. 1992. *Incremental Evolution of Neural Network Architectures for Adaptive Behaviour*. Technical report. Cognitive Science Research Paper CSRP256. Brighton BN1 9QH, UK.

Colorni, A., Dorigo, M. and Maniezzo, V. 1991. Distributed optimization by ant colonies. *In*: *Proceedings of ECAL91 – European Conference on Artificial Life*. Elsevier Publishing, Paris, pp. 134–42.

Cormen, T.H., Leiserson, C.E., Rivest, R.L., *et al.* 2003. *Introduction to Algorithms*. 2nd edn. McGraw-Hill Science/Engineering/Math, Boston, MA.

Correll, N. and Martinoli, A. 2006. Collective inspection of regular structures using a swarm of miniature robots. *In*: *Springer Tracts in Advanced Robotics: Proceedings of the Ninth International Symposium on Experimental Robotics (ISER-04)*. Springer, Berlin, pp. 375–85.

Couzin, I.D. 2009. Collective cognition in animal groups. *Trends in Cognitive Sciences* **13**:36–43.

Couzin, I.D., Krause, J., Franks, N.R., *et al.* 2005. Effective leadership and decision-making in animal groups on the move. *Nature* **433**(7025):513–6.

Daly, R. and Shen, Q. 2009. Learning Bayesian network equivalence classes with ant colony optimization. *Journal of Artificial Intelligence Research* **35**:391–447.

Deneubourg, A., Aron, S., Goss, S., *et al.* 1990a. The self-organizing exploratory pattern of the Argentine ant. *Journal of Insect Behavior* **3**:159–68.

Deneubourg, J.L., Goss, S., Franks, N., *et al.* 1990b. The dynamics of collective sorting robot-like ants and ant-like robots. *In*: *Proceedings of the First International Conference on Simulation of Adaptive Behavior on From Animals to Animats*. MIT Press, Cambridge, MA, pp. 356–63.

Diamond, J. 2006. *Collapse: How Societies Choose to Fail or Succeed*. Penguin, New York, NY.

Dorigo, M., Maniezzo, V. and Colorni, A. 1991. *The Ant System: An Autocatalytic Optimizing Process*. Technical report 91–016 Revised. Milan, Italy.

Dorigo, M., Trianni, V., Şahin, E., *et al.* 2004. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots* **17**:223–45.

Drexler, K.E. 1986. *Engines of Creation: The Coming Era of Nanotechnology*. Anchor, Harpswell, ME.

Eberhart, R. and Shi, Y. 2001. Particle swarm optimization: developments, applications and resources. *In*: *Proceedings of the 2001 Congress on Evolutionary Computation*. IEEE, New Jersey, pp. 81–6.

Eberhart, R.C. and Hu, X. 1999. Human tremor analysis using particle swarm optimization. *In*: *Proceedings of the Congress on Evolutionary Computation, Washington DC, USA*. IEEE, New Jersey, pp. 1927–30.

Eberhart, R.C., Shi, Y. and Kennedy, J. 2001. *Swarm Intelligence (The Morgan Kaufmann Series in Artificial Intelligence)*. 1st edn. Morgan Kaufmann, Waltham, MA.

Emerson, A.E. 1938. Termite nests – a study of the phylogeny of behavior. *Ecological Monographs* **8**:247–84.

Fortnow, L. and Homer, S. 2002. A short history of computational complexity. *In*: *The History of Mathematical Logic*. North Holland Publishing, Amsterdam.

Freitas, R.A. 1983. Terraforming Mars and Venus using machine self-replicating systems/SRS. *British Interplanetary Society, Journal (Interstellar Studies)* **36**:139–42.

Galehdar, A., Thiel, D.V., Lewis, A., *et al.* 2009. Multiobjective optimization for small meander wire dipole antennas in a fixed area using ant colony system. *International Journal of RF and Microwave Computer-Aided Engineering* **19**:592–7.

Garnier, S., Gautrais, J. and Theraulaz, G. 2007. The biological principles of swarm intelligence. *Swarm Intelligence* **1**:3–31.

Giles, J. 2004. Nanotech takes small step towards burying 'grey goo'. *Nature* **429** (6992):591.

Giles, J. 2005. Internet encyclopaedias go head to head. *Nature* **438**(7070):900–01.

Green, D.G., Leishman, T.G. and Sadedin, S. 2006. Dual phase evolution – a mechanism for self-organization in complex systems. *InterJournal*, 1–8.

Green, D.G., Leishman, T.G., Sadedin, S., *et al.* 2010. Of ants and men – the role of complexity in social change. *Evolutionary and Institutional Economics Review* **6**:259–75.

Halley, J.D. and Winkler, D.A. 2008. Critical-like self-organization and natural selection: two facets of a single evolutionary process? *Biosystems* **92**:148–58.

Handl, J. and Meyer, B. 2007. Ant-based and swarm-based clustering. *Swarm Intelligence* **1**:95–113.

Harada, K., Oetomo, D., Susilo, E., *et al.*. 2010. A reconfigurable modular robotic endoluminal surgical system: vision and preliminary results. *Robotica* **28**:171–83.

Heppner, F. and Grenander, U. 1990. A stochastic nonlinear model for coordinated bird flocks. *In*: Krasner, E. (Ed.), *The Ubiquity of Chaos*. AAAS Publications, Washington, DC, pp. 233–8.

Hölldobler, B. and Wilson, E.O. 1990. *The Ants*. Belknap Press, Harvard, MA.

Hornik, K., Stinchcombe, M. and White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks* **2**:359–66.

Hsiang, T.-R., Arkin, E.M., Bender, M.A., *et al.* 2003. Algorithms for rapidly dispersing robot swarms in unknown environments. *In*: *Springer Tracts in Advanced Robotics: Algorithmic Foundations of Robotics V*. Springer, Berlin, pp. 77–94.

Imre Friedmann, E. and Ocampo-Friedmann, R. 1995. A primitive cyanobacterium as pioneer microorganism for terraforming Mars. *Advances in Space Research* **15**:243–6.

Ito, M., Noda, K., Hoshino, Y., *et al.* 2006. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. *Neural Networks* **19**:323–37.

Johansson, C. and Lansner, A. 2007. Towards cortex sized artificial neural systems. *Neural Networks* **20**:48–61.

Juang, C.F. 2004. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE transactions on Systems, Man, and Cybernetics. Part B, Cybernetics* **34**:997–1006.

Juang, C.F., Hsiao, C.M. and Hsu, C.H. 2009. Hierarchical cluster-based multispecies particle-swarm optimization for fuzzy-system optimization. *IEEE Transactions on Fuzzy Systems* **18**:14–26.

Julian, G.E. and Cahan, S. 1999. Undertaking specialization in the desert leaf-cutter ant *Acromyrmex versicolor*. *Animal Behaviour* **58**:437–42.

Kassabalidis, I., Sharkawi, El M.A., Marks, R.J., *et al.* 2001 (November). Swarm Intelligence for Routing in Communication Networks. *In*: *Global Telecommunications Conference, 2001. GLOBECOM'01*. IEEE, New Jersey, pp. 3613–7.

Kauffman, S.A. 1993. *The Origins of Order: Self Organization and Selection in Evolution*. Oxford University Press, New York, NY.

Kennedy, J. and Eberhart, R. 1995 (August). Particle swarm optimization. *In*: *Proceedings of the IEEE International Conference on Neural Networks, 1995*, IEEE, New Jersey, vol. **4**, pp. 1942–8.

Knuth, D.E. 1997. *Art of Computer Programming, Volume 1: Fundamental Algorithms.* 3rd edn. Addison-Wesley Professional, Boston, MA.

Korb, J. 2003. Thermoregulation and ventilation of termite mounds. *Naturwissenschaften* **90**:212–9.

Krause, J. and Ruxton, G.D. 2002. *Living in Groups*. Oxford University Press, New York, NY.

Krause, J., Ruxton, G.D. and Krause, S. 2009. Swarm intelligence in animals and humans. *Trends in Ecology and Evolution* **25**:28–34.

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

Langville, A.N., Meyer, C.D. and Fernández, P. 2008. Google's PageRank and beyond: the science of search engine rankings. *The Mathematical Intelligencer* **30**:68–9.

Merkle, D., Middendorf, M. and Schmeck, H. 2002. Ant colony optimization for resource-constrained project scheduling. *IEEE Transactions on Evolutionary Computation* **6**:333–46.

Minsky, M.L. 1967. *Computation: Finite and Infinite Machines.* Prentice Hall, Englewood Cliffs, NJ.

Minsky, M.L. and Papert, S.A. 1987. *Perceptrons – Expanded Edition: An Introduction to Computational Geometry.* MIT Press, Cambridge, MA.

Muñoz Zavala, A., Hernández Aguirre, A. and Villa Diharce, E. 2005a. Particle evolutionary swarm optimization with linearly decreasing ε-tolerance. *In*: Gelbukh, A., Albornoz, Á. and Terashima-Marín, H. (Eds), *MICAI 2005: Advances in Artificial Intelligence*, vol. 3789. Springer, Berlin Heidelberg, pp. 641–51.

Muñoz Zavala, A.E., Aguirre, A.H. and Villa Diharce, E.R. 2005b. Constrained optimization via particle evolutionary swarm optimization algorithm (PESO). In: *GECCO'05: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation.* ACM, New York, NY, pp. 209–16.

Muñoz Zavala, A.E., Villa Diharce, E.R. and Hernández Aguirre, A. 2005c. Particle evolutionary swarm for design reliability optimization. *In*: *Evolutionary Multi-Criterion Optimization*. Springer, New York, NY, pp. 856–69.

Nicholson, J.A. 2005. Flash! mobs in the age of mobile connectivity. *Fibreculture Journal* **6**.

Papineau, D., Walker, J.J., Mojzsis, S.J., *et al.* 2005. Composition and structure of microbial communities from stromatolites of Hamelin Pool in Shark Bay, Western Australia. *Applied and Environmental Microbiology* **71**:4822.

Parpinelli, R.S., Lopes, H.S. and Freitas, A.A. 2002. Data mining with an ant colony optimization algorithm. *IEEE Transactions on Evolutionary Computation* **6**:321–32.

Perez, R. and Behdinan, K. 2007. Particle swarm approach for structural design optimization. *Computers and Structures* **85**:1579–88.

Pinto, P.C., Nägele, A., Dejori, M., Runkler, T.A., *et al.* 2009. Using a local discovery ant algorithm for Bayesian network structure learning. *IEEE Transactions on Evolutionary Computation*, **13**:767–79.

Pitts, W. and Mcculloch, W. 1947. How we know universals the perception of auditory and visual forms. *Bulletin of Mathematical Biology* **9**:127–47.

Poli, R., Kennedy, J. and Blackwell, T. 2007. Particle swarm optimization. *Swarm Intelligence* **1**:33–57.

Polymath, D.H.J. 2010. A new proof of the density Hales–Jewett theorem. *Arxiv preprint arXiv:0910.3926v2*.

Resnick, P., Kuwabara, K., Zeckhauser, R., *et al.* 2000. Reputation systems. *Communications of the ACM* **43**(12):45–8.

Reynolds, C.W. 1987. Flocks, herds, and schools: a distributed behavioral model. *Computer Graphics* **21**(4):25–34.

Ripley, B.D. 1996. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge.

Rojas, R. 1996. *Neural Networks: A Systematic Introduction*. Springer, New York, NY.

Rosenblatt, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review* **65**:386–408.

Rowe, J.E., Vose, M.D. and Wright, A.H. 2009. Reinterpreting no free lunch. *Evolutionary Computation* **17**:117–29.

Rubenstein, M. and Shen, W. 2009. Scalable self-assembly and self-repair in a collective of robots. *In*: *IROS IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, New Jersey, pp. 1484–9.

Şahin, E. 2005. From sources of inspiration to domains of application. *In*: Sahin, E. and Spears, W. (eds), *Lecture Notes in Computer Science: Swarm Robotics*, Springer, Berlin, pp. 10–20.

Schumacher, C., Vose, M.D. and Whitley, L.D. 2001. The no free lunch and problem description length. *In*: Spector, L., Goodman, E.D., Wu, A., Langdon, W.B., Voigt, H.M., Gen, M., *et al.* (Eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*. Morgan Kaufmann, San Francisco, CA, pp. 565–70.

Shen, W.M., Will, P., Galstyan, A., *et al.* 2004. Hormone-inspired self-organization and distributed control of robotic swarms. *Autonomous Robots* **17**:93–105.

Sipser, M. 2005. *Introduction to the Theory of Computation*. 2nd edn. Course Technology, Kentucky.

Spears, W.M., Spears, D.F., Hamann, J.C., *et al.* 2004. Distributed, physics-based control of swarms of vehicles. *Autonomous Robots* **17**:137–62.

Su, M.-C. and Chang, H.-T. 2001. A new model of self-organizing neural networks and its application in data projection. *IEEE Transactions on Neural Networks* **12**:153–8.

Trianni, V., Labella, T.H. and Dorigo, M. 2004. Evolution of direct communication for a swarm-bot performing hole avoidance. *In*: *Ant Colony, Optimization and Swarm Intelligence*. Springer, Berlin, pp. 174–97.

Turgut, A., Çelikkanat, H., Gökçe, F., *et al.* 2008. Self-organized flocking in mobile robot swarms. *Swarm Intelligence* **2**:97–120.

Wachowiak, M.P., Smolikova, R., Zheng, Y., *et al.* 2004. An approach to multimodal biomedical image registration utilizing particle swarm optimization. *IEEE Transactions on Evolutionary Computation* **8**:289–301.

Werfel, J., Yam, Y.B. and Nagpal, R. 2005. Building patterned structures with robot swarms. *In*: *IJCAI'05: Proceedings of the 19th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, San Francisco, CA, pp. 1495–502.

Whitley, D. and Watson, J. 2005. Complexity theory and the no free lunch theorem. In: Burke, E.K. and Kendall, G. (eds), *Search Methodologies*. Springer US, New York, NY, pp. 317–39.

Whitley, D. and Rowe, J. 2008. Focused no free lunch theorems. *In*: *GECCO'08: Proceedings of the 10th annual conference on Genetic and evolutionary computation*. ACM, New York, NY, pp. 811–8.

Winfield, A.F.T. and Nembrini, J. 2006. Safety in numbers: fault-tolerance in robot swarms. *International Journal of Modelling, Identification and Control* **1**:30–7.

Winfield, A.F.T., Sa, J., Gago, M.C.F., *et al.* 2005. On formal specification of emergent behaviours in swarm robotic systems. *International Journal of Advanced Robotic Systems* **2**:363–70.

Wolpert, D.H. and Macready, W.G. 1995. *No Free Lunch Theorems for Search*. Technical report SFI-TR-95–02–010. Santa Fe, NM.

Wolpert, D.H. and Macready, W.G. 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation* **1**:67–82.

Wu, Z.-Y. and Song, H.-T. 2008. Ant-based energy-aware disjoint multipath routing algorithm for MANETs. *The Computer Journal*, February, bxn007+.

Xiong, W. and Wang, C. 2009. A hybrid improved ant colony optimization and random forests feature selection method for microarray data. *In*: *International Conference on Networked Computing and Advanced Information Management*. IEEE, New Jersey, pp. 559–63.

Yao, X. 1993. Evolutionary artificial neural networks. *International Journal of Intelligent Systems* **4**:539–67.

Yoshida, H., Kawata, K., Fukuyama, Y., *et al.* 2000. A particle swarm optimization for reactive power and voltage control considering voltage security assessment. *IEEE Transactions on Power Systems* **15**:1232–9.

Zaka, I., Rehman, H., Naeem, M., Shah, S. and Ahmad, J. 2008. Narrowband jammer excision in CDMA using particle swarm optimization. *In*: Huang, D.-S., Wunsch, D.C., Levine, D.S. and Jo, K.-H. (Eds), *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues*. Lecture Notes in Computer Science, vol. 5226. Springer, Berlin, Heidelberg, pp. 601–09.

Zhang, G., Patuwo, B.E. and Hu, M.Y. 1998. Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting* **14**:35–62.