Chapter 2
Self-organising Systems

Giovanna Di Marzo Serugendo, Marie-Pierre Gleizes, and Anthony Karageorgos

Self-organisation and emergence in software systems.

Objectives The objective of this chapter is to introduce the subject of self-organising software systems. When you will have read this chapter, you will:

- Know the answers to key introductory questions to self-organising systems;
- Understand what self-organising software is and why it is important;
- Obtain an understanding of different self-organising system types that can be found in nature;
- Obtain an overview of various self-organising applications.

2.1 Self-organising Systems: An Overview

The study of self-organising systems is a field that has been explored at least since 1953 with the work done by Grassé [43], who studied the behaviour of insect so-

G. Di Marzo Serugendo
Birkbeck College, University of London, London, UK
e-mail: dimarzo@dcs.bbk.ac.uk

M.-P. Gleizes
IRIT, Université Paul Sabatier, Toulouse, France
e-mail: Marie-Pierre.Gleizes@irit.fr

A. Karageorgos (✉)
Technological Educational Institute of Larissa, Larissa, Greece
e-mail: karageorgos@computer.org

G. Di Marzo Serugendo et al. (eds.), *Self-organising Software*,
Natural Computing Series,
DOI 10.1007/978-3-642-17348-6_2,

cieties. Grassé found that these societies show changing forms of order occurring without any central point of control.

In other works it has been found that many existing systems demonstrate self-organisation, such as planetary systems, organic cells, living organisms and animal societies. All these systems exhibit recurrent properties inherent to self-organisation, and are therefore termed *self-organising systems*. Self-organising systems are encountered in many scientific areas including biology, chemistry, geology, sociology and information technology, and considerable research has so far been undertaken to study them.

A large number of artificial self-organising systems are designed based on self-organisation mechanisms inspired by nature. Furthermore, recent research has been oriented towards introducing self-organisation mechanisms specifically for software applications, as well as entire software development techniques supporting self-organisation [33]. This trend originates from the fact that current software applications need to cope with requirements and constraints stemming from the increased dynamism, sophisticated resource control, autonomy and decentralisation inherent in contemporary business and social environments. The majority of these characteristics and constraints are the same as those which can be observed in natural systems exhibiting self-organisation.

The variety of systems where the notion of self-organisation is found makes it difficult to find a precise definition of what self-organisation is. In its simplest form, self-organisation can be considered as the autonomous arrangement of parts of a system in such a way as to be non-random. A more detailed discussion on this topic is provided in Chap. 3.

In the following we provide a brief description of the notions of self-organisation and emergence, and we outline the main types of self-organising systems. In all systems, an important issue is their capacity to deploy the effective global behaviour that permits the realisation of their intentional or non-intentional goals.

2.1.1 Self-organisation and Emergence

Intuitively, self-organisation refers to the fact that a system's structure or organisation appears without any explicit control or constraints imposed from *outside* the system. In other words, the organisation is intrinsic to the self-organising system, and it results from *internal* constraints and mechanisms, which are based on *local interactions* between its components [24]. These interactions are often *indirect* and are carried out through the environment [43]. The non-deterministic and dynamic nature of interactions causes *emergent* system properties to appear, which transcend the properties of all individual sub-units of the system [49]. Furthermore, the dynamic operations affect and modify the *environment* the system is situated in, and in turn alterations to the environment influence the system again in a feedback loop [68]. In most cases environmental influences and perturbations do not affect the internal mechanisms which cause dynamic re-organisation. As a result, the system *evolves dynamically* [17] either in time or space, and it can either aim to maintain a stable form or show transient phenomena.

Generally, the emergent phenomena is an externally identifiable outcome, for example a particular pattern or structure, property, behaviour or system state, which, although not explicitly represented at a lower level, appears at a higher level. That complex collective behaviour usually occurs without *any central control*, and it is derived from the simple local individual behaviours and interactions. One well-known example of emergent system behaviour is the collective behaviour shown in a colony of ants which sort their eggs without having any particular ant knowing and centrally applying some sorting algorithm.

Self-organisation and emergence are not perfect; units in a self-organising system are prone to opposing actions, their behaviour may induce needless redundancies, and decentralised control limits the ability of the system to find the globally optimal solution [84]. However, for systems that are complex and operate in a dynamic environment, the use of self-organisation offers significant advantages such as increased scalability and robustness and reduced communication and unit processing costs [66].

The body of work in the area of self-organising systems and emergent phenomena follows different streams. There is a large body of work dedicated to the study of natural systems to understand their underlying mechanisms (e.g. understanding natural swarms, human organisations, earthquakes, etc.). Self-organisation and emergent phenomena are also studied from a theoretical and generic perspective in the field of *Complex Adaptive Systems* [11]. Alternatively, the mechanisms or theoretical concepts identified during the study of self-organising systems provide nature-inspired techniques for engineering artificial systems in many scientific areas, such as Computational Biology [25], Synthetic Biology [36], Artificial Life [19], Robotics [78], Computer Science [34].

This book concentrates on the latter, exploiting techniques inspired from natural self-organising systems in ICT-related applications, such as optimisation problem solving, P2P protocols, sensor networks or access control.

2.1.2 Natural Systems

Self-organising and emergent phenomena can be observed in many natural systems. For example, insects that live in colonies, such as ants, bees, wasps and termites, have been shown to seamlessly integrate their individual activities, while every single insect seems to operate individually and without any central supervision. Interaction among members in such colonies is carried out indirectly by asynchronously modifying and subsequently perceiving the state of their environment. Other examples of natural self-organising systems include flocks of birds and schools of fish. In these cases, collective self-organising behaviour is achieved by synchronous perception of the environment and by following simple rules. For example, by aiming to remain in close proximity while avoiding collision with similar group members

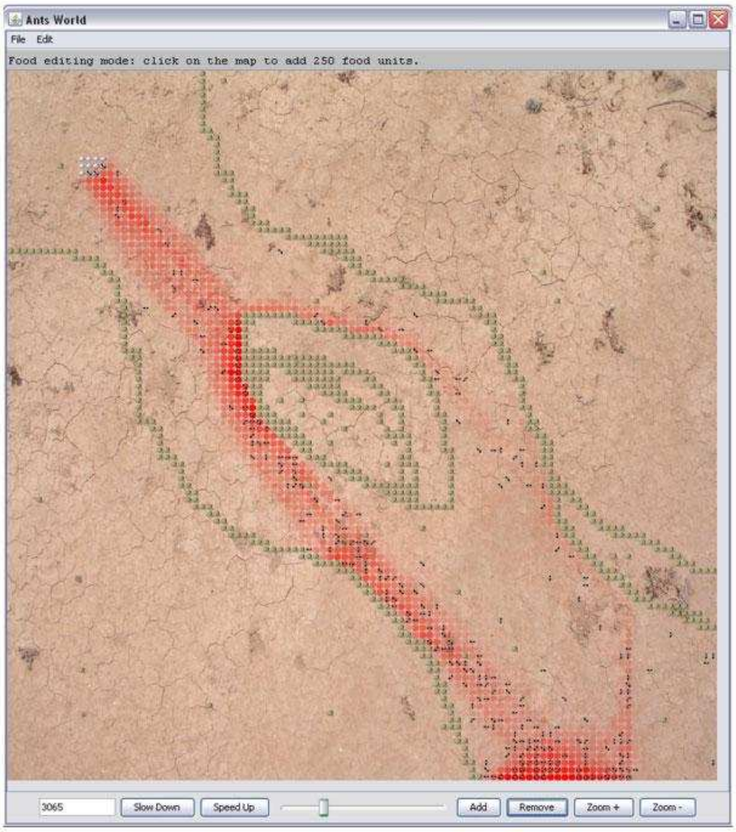

Fig. 2.1 Path emerging from foraging ants

and at the same time by attempting to keep a distance from dissimilar entities, birds or fish significantly reduce the probability of being attacked by a predator [30].

Such systems exhibit many interesting complex behaviours, and they have emergent properties resulting from local interactions between elementary behaviours exercised individually. The emergent collective behaviour is the outcome of self-organisation processes in which members are engaged through their repeated actions and interactions with their evolving environment [49].

A typical example of self-organisation and emergence in natural systems concerns foraging ant colonies [31]. Foraging ants explore their environment seeking food. When they succeed in their task, they return to the nest after placing in the environment chemical substances which are subsequently used by other ants as indicators of proximity to food source. Not long after the food source is discovered, a path consisting of dropped chemical substances is created between the food source and the nest; actually, it is the shortest possible path given the environmental constraints, and it emerges as a result of the collective activity of ants (see Fig. 2.1 for a simulation view). That path can be identified only by external observers; for example, the individual ants cannot see it as a whole.

The field of research that investigates models and computational techniques inspired by nature is termed *Natural Computing*, and it attempts to understand systems found in nature in terms of information processing. It is a highly interdisciplinary field that connects the natural sciences with computing science, both at the level of information technology and at the level of fundamental research. Natural computing covers many areas including pure theoretical research, algorithms and software applications, as well as biology, chemistry and physics experimental laboratory research [55].

Natural systems can be broadly classified as Physical Systems, Biological Systems and Social Systems. A brief description of each category is provided in the next sections, and more details are provided in Chap. 4.

2.1.2.1 Physical Systems

Theories of self-organisation were 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 [45, 76]. A common characteristic of physical self-organising systems is the existence of some *critical threshold*, which causes an immediate change to system state when reached. That critical threshold can be a combination of values of certain system variables, for example temperature, pressure and speed. The self-organisation effect is observed *globally* when the system transits from a chaotic disordered state to a stable one. For example, in a thermodynamic system, such as one consisting of a gas quantity, the system properties temperature and pressure are emergent since they are not determined from any particular gas particle. Instead, they are defined by the positions and velocities of all particles in the system.

Similarly, chemical reactions create new molecules that have properties that none of the atoms exhibit before the reaction takes place [11]. Moreover, the magnetisation of a multitude of spins in magnetic materials[1] is a clear case of self-organisation because, under a certain temperature, the magnetic spins spontaneously rearrange themselves pointing all in the same direction and creating thus a strong emerging magnetic field [48]. Additional physical and chemical systems are described in Chaps. 3 and 4.

2.1.2.2 Biological Systems

Self-organisation is a common phenomenon in subsystems of living organisms. As a result, an important field in biological research is the determination of invariants in the evolution of living organisms and in particular the spontaneous appearance of order in living complex systems due to self-organisation. In the optic of biological research, the common meaning of self-organisation is defined by the global emergence of a particular behaviour or feature that cannot be reduced to the properties of individual system's components such as molecules, agents and cells [24].

An example described in [75] is the self-organisation of the cytoskeleton due to collective processes of reaction and diffusion of the cytoskeletal filaments. The cytoskeleton is the basis of the internal architecture of the cytoplasm of eukaryotic cells. Eukaryotic cells are cells belonging to higher-level organisms which contain

[1] A spin is a tiny magnet.

a true nucleus bounded by a nuclear membrane. Such cells are found in plants and animals, and they are commonly organised into organs, such as the liver, or subsystems, such as the nervous system, which, by means of metabolic processes, despite the increase of system complexity, remarkably provide higher-level richer functionality. That resulting functionality is completely new, and it transcends all individual functionalities offered by the respective constitutive cells. For instance, human body subsystems transparently manage vital functions, such as blood pressure, digestion, or antibodies creation.

Eukaryotic cells are the fundamental functioning units of life, and they evolve progressively according to external changes to their environment. Evolution is achieved with the use of internal metabolic processes such as mutation or natural selection, and it provides the basis for the evolution observed in living organisms and natural species.

Additional examples of self-organising biological systems include the immune system of mammalians, the regeneration of cells and the human brain behaviour. A typical example of the latter is the apparition of conscience in humans. Conscience is viewed by Searle [83] as a property of the brain at the higher or global level. Biologically, the brain is a complex system composed of a set of neurons and interactions between them. Although conscience is a result of neuron operations done at a lower level, it is currently not possible to understand or explain human conscience by observing the brain neurons and their interactions.

2.1.2.3 Social Systems

In natural system's societies, entities commonly exhibit social behaviours leading to self-organisation, self-adaptation and self-maintenance of the society organisation, which often is observed in the form of some global societal behaviour. Individual social behaviours range from those observed in biological entities, for example bacteria, cells and insects, such as spiders, to those observed in larger animals and humans. One important characteristic of global societal behaviours is their formation as results developed from relatively simple interactions in a network of individuals. These resulting behaviours are considered to be driven by dynamic processes that are governed by simple but generic laws [12, 37].

For example, an important reason that has historically triggered collective behaviour in natural societies is the ultimate goal of species survival. This goal is not explicitly expressed at the individual level, but it is reflected in the collective behaviour of society members towards the emergence of *social functions* and group dynamics allowing the maintenance of the system's organisation. For instance, social insects organise themselves to perform activities such as food foraging or nest building. Communication among insects is realised by an indirect communication mechanism, termed *stigmergy*, which is implemented through the society environment. Insects, such as ants, termites or bees, implement stigmergy by marking their environment using a chemical volatile substance, termed *pheromone*,[2] and subsequently arrange the direction of their movements based on the location of pheromones in their environment, for example, as ants do to mark a food trail. The pheromonal information constitutes an indirect communication means through the insects' environment. Insects generally have a simple behaviour, at least as far as the collective outcome is concerned [84], and none of them alone can provide the complete solution to the problem; for example, no ant is aware of the exact location of the food source found. However, the interactions between individual behaviours give rise to an organised society collective behaviour, for example ants are able to explore their environment, find food and efficiently inform the rest of the colony.

Apart from other living organisms, humans commonly organise into advanced societies and organisations which can serve many purposes. For example, humans can use direct communication, engage in negotiations, build whole international economies and organise stock markets. Social behaviour of humans is typically self-organising, and it normally gives rise to emergent complex global behaviours. In many cases, individual human behaviour is based on small-range local information, and communication is carried out on local direct or indirect interactions which produce complex societal behaviours. A representative example of an emergent phenomenon in human societies is that of common beliefs, for example work ethics, developed and fostered through local communications such as gossip.

2.1.3 Business and Economics Systems

We distinguish natural systems from business and economic systems, since generic laws guiding self-organisation in the former type of systems are dictated by nature, whereas in the latter, self-organisation is governed by business and market laws.

In business and economic systems, individual behaviours are goal-oriented, and their primary goal is to increase their profit. In this case, the system's dynamics is handled by the activity developed to face business and economic constraints to reach a global equilibrium through which the system can survive. In this context, various models governing business operations, such as management, marketing and economic models, which are based on self-organisation have been developed.

For example, one such model from the marketing domain is the *one-to-one variable pricing* model [46] which refers to providing an individual offer to each customer using Internet technologies. The model involves self-organisation of the marketing policies by changing customers targeted and the prices quoted based on market dynamics, customer characteristics and the business goals. An example of a company using this type of marketing is FedEx [101]. The company allows customers to access computer systems, via the Web site, to monitor the status of their packages. For corporate customers, FedEx provides software tools that enable the

[2] Pheromones are chemical substances deposited by insects in their environment at regular intervals so that other insects sense and follow them.

organisation to automate shipping and track packages using their own computing resource. Each customer is offered different prices depending on a variety of parameters. Many websites, such as eBay, also apply variable pricing for their offers.

Another example of a self-organising business model from the area of management is the *theory of activity* described in [100]. According to that model, a business is viewed as consisting of networks of working groups that can change their structure, links and behaviour in response to business requirements. The aim of that model is to balance the self-organisation decisions that are both taken by managers and shaped from interactions between employees during business operations. The emphasis of the theory of activity is on solving potential conflicts of interests both of the inner and outside co-operative activity of the company.

2.1.4 Artificial Systems

Artificial self-organising systems are those built from the beginning with embedded self-organisation capabilities. Mechanisms can be borrowed from existing natural systems, or they can be created explicitly for that purpose.

Among artificial self-organising systems, we observe different trends ranging from application of naturally-inspired self-organising models to the establishment of new mechanisms and whole infrastructures supporting self-organisation of artificial systems. Swarms provide a great source of inspiration, especially for fixed and mobile networks systems management [20], such as routing, load balancing [74], or security [41]. Holarchies as well have inspired researchers dealing with e-Government and e-Society issues [93]. At the level of whole infrastructures (middleware) supporting artificial self-organising systems, some works take their inspiration from magnetic fields [65] or ants [10].

Furthermore, self-organising models aimed specifically for particular applications have been developed. For example, an artificially engineered self-organising system concerning coordination and organisation of a group of robots is reported in [80]. In that context, robots transport objects between two rooms connected with two corridors which cannot be used by more than one robots at a time. Robots have local perception and apply a cooperative attitude in selecting which corridor to use. The result is system performance improvement arising from corridor specialisation where movement in each corridor is done in a specific direction.

2.2 Self-organising Applications

Self-organisation is increasingly used in software applications to provide the solution to problems of various types. This is leveraged by the distributed nature of contemporary software, the highly changing environments, the large numbers of heterogeneous components software systems commonly consist of, and the difficulty in imposing central control in distributed software.

In the following we present an overview of the most common applications of self-organisation in software systems.

2.2.1 Multi-Agent Systems

An agent is a physical (robot) or a virtual (software) entity situated in an environment that changes over time: the physical world or an operating system respectively. Through its sensors, the agent is capable of perceiving its environment, and through its effectors, it is capable of performing actions that affect the environment. For instance, a robot may take notice of obstacles with an embedded camera and remove them from its way with an articulated arm. A software agent may understand a user's request through a user's interface and send an e-mail to the user once the request has been satisfied [54, 81].

Every single agent has one or more limitations, which can be categorised into cognitive limitations, physical limitations, temporal limitations and institutional limitations. Cognitive limitations resemble the fact that individuals are rationally bounded. It means that the data, information, and knowledge an individual can process and the detail of control an individual can handle is limited. As tasks grow larger and more complex, techniques must be applied to limit the increase of information and the complexity of control. Individuals can be limited physically, because of their physiology or because of the resources available to them. Temporal limitations exist where the achievement of individual goals exceeds the lifetime of an individual, or the time over which resources are available for achieving a goal. Finally, individuals can be legally or politically limited.

To overcome their limitations, agents group together and form *multi-agent systems*, or societies of agents, where they work together to solve problems that are beyond their individual capabilities [77]. A robot has to bring a cup from the office to the kitchen. It may ask the need of another agent to bring the cup in the kitchen if it is itself unable to reach the kitchen whose door is closed. On the basis of its knowledge about the user's behaviour, an assistant agent may decide to regularly inform that user about new computer science books, without the user having explicitly notified the agent to do that. In order to obtain this information, the agent may need to contact other agents aware of computer science books.

Agents interact (communicate, coordinate, negotiate) with each other and with their environment. Interactions among agents are normally governed by some coordination model. An agent can communicate directly or indirectly with other agents for cooperation or competition purposes. Since the agent perceives its environment and interacts with other agents, it is able to build a partial representation of its environment, which constitutes its knowledge. Usually, in a multi-agent system, interactions are not pre-defined, and there is no global system goal. The interaction dynamics between an agent and its environment lead to emergent structures or emergent functionality, even though no component is responsible for producing a global goal.

More details about the concepts of agent and multi-agent systems are provided in Chap. 5.

2.2.2 Computational Grids and P2P Systems

Computational Grids provide the software and networking infrastructure required to integrate computational engines, scientific instruments, data repositories and human expertise to solve a single large problem generally in science and engineering domains. Grid computing essentially refers to any distributed cluster of compute resources that provides an environment for the sharing and managing of the resource and for the distribution of tasks based on configurable service-level policies.

Current resources being provided within Computational Grids are owned by national or regional centres, or by research institutions, and therefore access rights and usage need to be pre-defined and approved. However, as resources from less trusted users are provided, the need to organise them into dynamic communities, based on a number of different criteria, such as performance, trust or cost of ownership and usage, becomes significant. Self-organisation therefore plays an important role in identifying how such communities can be formed and subsequently disbanded. For example, a utility-based approach for forming such communities is explored in [64].

Other issues relate to resource selection and management, such as the distribution of tasks to the best available node and the distribution and management of data between compute nodes, while optimising global performances. Solutions vary from the use of mobile agents for resource and task distribution [51] to human market inspired techniques for adaptively changing the application placement and workload assignment to satisfy the dynamic applications workloads [62], to self-adaptive techniques based on monitoring of processors and computation of efficiency to automatically adapt the number of processors or migrate computation away from overloaded resources [103], to techniques inspired by bee foraging behaviour [59] allowing grid clients to dynamically select the most appropriate algorithms for executing chunks of data.

In parallel to computational grids, there has been an emerging interest in trying to integrate resources across organisational boundaries through file or CPU sharing software (such as KaZaA [3] and Gnutella [2] for file sharing and Entropia [1] and UD [4] for CPU sharing). These individual resources are often geographically distributed and may be owned by different administrators (or exist within different independently administered domains). Self-organisation in P2P and MANET encompass gossip-based overlay topology management [53] or decentralised techniques for routing, updates and identity maintenance [8].

2.2.3 Service Emergence

In many cases, the service provided by some applications emerges from joint provision and interaction between constituent subservices. For example, Itao et al. [52] propose Jack-in-the-Net (Ja-Net), a biologically-inspired approach to design emergent network applications and services in large-scale networks. In Ja-Net, network applications and services are dynamically created from local interactions and collaboration of self-organising entities, called cyber-entities. Each cyber-entity implements a simple functional component of the overall service or application. Furthermore, it follows simple behavioural rules, similar to the behaviours of biological entities, such as energy exchange with the environment, migration or replication, and relationship establishment. Services emerge from strong relationships among cyber-entities. Indeed, cyber-entities record information about peer cyber-entities during a relationship. Once relationships among a collection of cyber-entities are strong enough, they form a group and create a service. Relationship strengths are evaluated on the basis of the utility degree of each cyber-entity participating in the service. The utility degree is estimated using user feedback on the delivered service. In addition to service emergence, the system exhibits a natural selection mechanism based on the notions of energy stored or spent by cyber-entities and the diversity of services created. Due to the migration, replication and possible deaths of entities, the system is also able to adapt to networks changes.

Along the same lines, with self-adaptability and evolvability in mind, Viroli et al. [98] propose to engineer ecosystems of services getting inspiration from natural ecosystems. Services and data sources are seen as autonomous individuals, and rules governing the ecosystem are inspired by those found in natural ecosystems, e.g. biochemical mechanisms.

2.2.4 Dynamic Web Page Communities

Flake et al. [39] show that Web pages self-organise into communities identified by pages connectivity. A Web page is a member of a community if it has more hyperlinks within the community than outside it. Flake et al. have defined a specific algorithm which highlights communities of related Web pages, on the basis of hyperlinks contained in the pages. From a self-organisation point of view, authors of Web pages independently put them on the Web with hyperlinks on other pages. Inserting a page on the Web modifies the environment, for example the world space of the Web pages, and this in turn modifies the behaviour of other authors of pages. It becomes then possible to reach existing pages from a new location as it becomes possible to reference and go through these new pages. By analogy with the ants metaphor, authors place Web pages (pheromone) on the Web (food trail). These Web pages contain specific information for other authors, who will reinforce (or not) the strengths among Web pages by referencing them. Authors then collectively but independently organise Web pages into communities.

2.2.5 Network Coordination

Since the early days of swarm intelligence research it has been argued that insect-like behaviour of simple entities working in groups could provide a powerful metaphor enabling the development of completely decentralised systems. A notable advantage of such systems was emphasised to be the collaboration of individual entities to produce collective and more complex global emergent behaviours [18].

These advantages have inspired computer and telecommunications network scientists to mimic insect foraging behaviour on the coordination and control of data network traffic [50].

A representative example is the ant-like approach suggested by Di Caro and Dorigo in [32], which uses swarm-based models and algorithms to solve network coordination problems. The hypothesis is that ant-based coordination would be able to better cope with network dynamics than direct human supervision. The proposed approach involves introducing an artificial ant network where each node periodically launches artificial ants assigned with the task of finding the route to a given destination. By simply sensing the intensity of the artificial pheromones located along the neighbour paths of the node, the artificial ants generate a map that provides the shortest route to any destination. In case of congestions, it was shown that this approach outperformed all other popular, at the time, network routing algorithms in terms of time needed to identify routes that would avoid traffic jams.

Additional examples of applications of nature-inspired algorithms to network coordination include applying ant-like behaviours to optimise network traffic [9, 16] and mimicking the division of labour and task allocation met in insect societies to optimise load balancing in network systems [82].

More details on bio-inspired swarm algorithms are provided in Chap. 4.

2.2.6 Network Security

The use of Mobile Agents in sophisticated applications offers advantages for constructing flexible and adaptable wide-area distributed systems. Notably, applications such as Intrusion Detection Systems (IDSs) and Intrusion Response Systems (IRSs) have become even more relevant in the context of large-scale network infrastructures, where traditional security mechanisms demonstrate severe weaknesses [67]. Indeed, as they can be retracted, dispatched, cloned or put in stand-by, mobile agents have the ability to sense network conditions and to load dynamically new functionality into a remote network node (such as a router). The network awareness of mobile agents can also significantly contribute to the detection of intrusions and enables providing appropriate response. Deriving from this trend, mobile agents have been proposed as support for Intrusion Detection (ID) and Intrusion Response (IR) in computer networks. The originality of this approach lies on the design of the intrusion detection and response system (IDRS). Indeed, the organisation of mobile agents follows the behaviour of natural systems to detect an intrusion as well as to answer an intrusion [41]. Schematically, there are two natural paradigms that have been referred to. Firstly, the *human immune system*, because the IDS is based upon principles derived from the immune system model, where Intrusion Detection Agents (ID Agents) map the functionality of the natural immune system to distinguish between normal and abnormal events (respectively “self” and “non-self” in the immune system) as explained in [40]. Secondly, the social insect *stigmergy* paradigm, because the IRS is based upon principles derived from this paradigm. In fact, Intrusion Response Agents (IR Agents) map the collective behaviour of an ant population by following a synthesised electronic pheromone specific to the detected intrusion until the source of the attack—in order to perform its response task. This pheromone has been previously diffused throughout the network by an ID Agent when it detected the attack. This kind of collective paradigm is very interesting because it consists in having each ant execute a rather light task (mobile agents play the role of ants in the IR System) to induce collectively a more complex behaviour. This approach is also very powerful because the ID System, as well as the IR System, are completely distributed in the network without any centralised control: both systems are essentially constituted by mobile agents which travel across the network, dynamically adjusting their routes according to collected events, without any simple way to trace them. Furthermore, mobile agents are quite polyvalent because they can detect and respond to intrusion. This enhances the difficulty for an attacker to distinguish between ID Agents and IR Agents. The use of AIS for addressing security has also been considered in MANET routing [69, 99].

More generally, de Castro and Timmis [26] propose a framework for engineering artificial immune systems. They provide a detailed review of the biological immune system and several artificial ones from different domains.

More details on mobile agents are given in Chap. 5. Network security and the human immune system are further discussed in Chap. 16.

2.2.7 Bio-inspired Robot Teams

Researchers have also been inspired by living systems to build robots. Lots of recent researches in robotics use insect-based technology where robots self-organise to accomplish a task (gathering a set of objects at a precise location for instance). As it is the case for ants' populations, the populations of robots own a local view of their environment, they can share individual information with other robots and co-operate with them. One direct application of the self-organisation with robots is the building of a global cartography of the environment where they are immersed without having each the knowledge of the global topology of the environment. In the approach described in [97] the robots' perception of their environment is tightly connected to their action, similarly to many successful biological systems. Robots perceive their environment locally by sending a simple visual information to their control system. The overall behaviour of the system emerges from the coordination, integration and competition between these various visual behaviours. Other examples include robot-based exploration [88], swarms of robots assembling together to create an organism similar to a multi-cell robots organism [7, 61, 63], swarms of homogeneous self-assembling robots to overcome obstacles or to transport object [5, 44], and heterogeneous cooperating robots [6].

2.2.8 Manufacturing Control

The self-organising behaviour exhibited in insect colonies has often been utilised in agent-based software aimed for manufacturing control [22, 23, 29, 56, 71, 89, 90]. A representative example of this approach is the PROSA[3] system described in [95, 96]. PROSA is an agent-based reference software architecture, where agents are modelled as artificial ants, and the whole system emulates an ant colony. There are three basic types of agents, namely *product agents*, *order agents* and *resource agents*. Each agent type is responsible for one aspect of manufacturing control, respectively: (i) recipes or process plans, (ii) internal logistics and (iii) resource handling. These basic agents are structured using object-oriented concepts like aggregation and specialisation. *Staff agents* can be added to assist the basic agents with expert knowledge.

Product agents own a 'product model' of a product type, and they act as information servers to other agents, delivering the right recipes to the right place. Order agents represent tasks. They are responsible for performing the corresponding work correctly and on time. They manage the physical products being produced, the product state model, and all logistic information processing related to the manufacturing operation. Resource agents provide a reflection of the underlying production system in the world of agents. They offer spaces for other agents to navigate through, to percept their neighbours and to store information that can observed by other agents and decay after a specified time, in a similar manner as the pheromones are deposited, sensed by ants and evaporate in the physical world.

Virtual agents (for example order agents or other task agents created for a particular purpose) acting as artificial ants move through this reflection of the physical world and collect information, which they make available elsewhere. Firstly, these ants collect information about the available processes, travel upstream and place the collected information at routing points. Secondly, exploring ants explore possible routings for the products being made, make a selection and propagate the resulting intentions through the 'reflection' (see Fig. 2.2). Resource agents receive this information about the intentions of their users and compile short-term forecasts for themselves. These forecasts allow up-to-date predictions of processing times used by the ants exploring routes and propagating intentions. All these activities are subjected to an evaporation (time-out) and refresh process that enables the system to keep functioning in a highly dynamic environments (frequent changes and disturbances).

An important aspect of the PROSA model is that agents can be aggregates consisting of multiple agents (see Chap. 11 for more details on holonic systems). In addition, agents can create new agents that virtually travel through the manufacturing system to create and maintain the dissipative fields that coordinate the behavior of the individual agents.

Another example where self-organisation is used for manufacturing control is described in [13]. The approach followed is termed *Bucket Brigades*, and it aims

[3] PROSA stands for Product-Resource-Order-Staff Architecture.

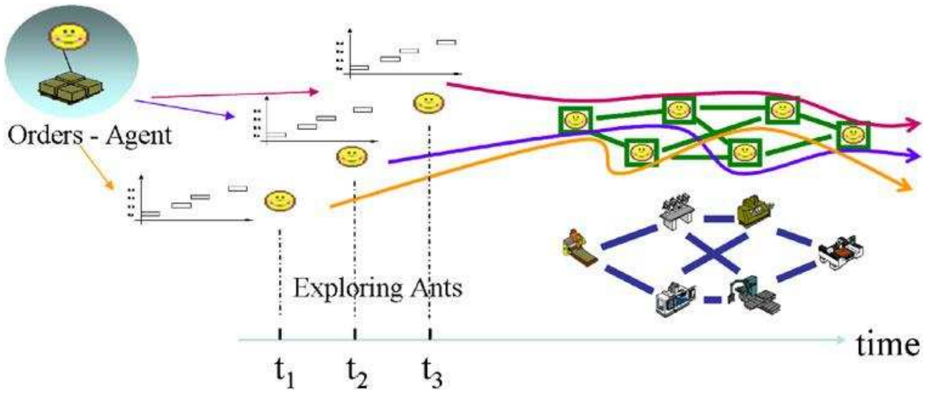

Fig. 2.2 Order agents and exploring ants in the PROSA approach

to optimise the work allocation in manufacturing systems. To this purpose, allocation of work is constantly updated by self-organising acts, each time improving load balancing and system performance. The important advantage of the achieved self-balancing work allocation is that there is no need for explicit supervision for measuring and reallocating work when needed. Furthermore, bucket brigades reduce the need for planning and management since they require no work-content model to share work effectively.

Bucket brigades have been applied in various sectors of manufacturing including assembly lines [14] and distribution processes [15]. An intuitive example discussed in [13] uses bucket brigades to mitigate time and motion expenses in an assembly line by enabling it to re-balance itself when needed.

The application of the bucket brigades concept on assembly lines is simple: each worker on the assembly line carries a product towards completion. However, when the last worker on the line finishes the product, he sends it off and then walks upstream to take over the work of his predecessor, who in turn walks back to take over the work of his predecessor and so on. (No overtaking is allowed.) Finally, after relinquishing his product, the first worker walks back to its starting position to begin a new product. When workers are sequenced from slowest to fastest (along direction of product flow), then it can be shown empirically, and proven mathematically, that workers spontaneously gravitate to the optimal division of work so that throughput is maximised.

2.2.9 Self-organising Sensor Networks

Self-organising wireless sensor networks are used for civil and military applications, such as volcanoes, earthquakes monitoring and chemical pollution checking. Sensor networks consist of self-organised nodes, which dynamically need to set up an ad hoc P2P network, once they are deployed in a given area. They need as well to calibrate themselves in order to adapt to their environment [102]. Sensor networks benefit also from recent technology enabling integration of a complete sensor system into small-size packages, as for instance the millimetre-scaled motes provided by the SmartDust project [86]. Another representative example is the STAFF real-time simulator [42] which uses an adaptive model for flood forecast composed of two levels of self-organising multi-agent systems (representing sensor nodes). The goal of each upper level agent is to compute the water level variation during a unitary period (typically an hour) and uses for that a weighted sum of agents in the lower level. The model's adaptive nature is obtained by adjustment of these weights, decided from the co-operation between the agents. It makes the model generic and improves its performances. Recent works on self-organising sensor networks focus on routing, synchronisation, power conservation [73] or decentralised collaborative detection of events [38].

2.2.10 Workflow Management

Workflow Management Systems (WfMSs) are software applications supporting the specification, design and enactment of business processes. In the global economy, businesses are constantly changing their structure and processes according to market dynamics. Therefore WfMSs need to be able to self-organise, namely adapt their functionality to support changes needed in business operations [21]. The majority of self-organising WfMSs are based on the agent metaphor. Agents provide flexibility and adaptability, and therefore they are particularly suitable for realising self-organising infrastructures.

Self-organising WfMSs implement self-organisation in three broad ways: by viewing business as Complex Adaptive Systems and applying relevant theories and models, by viewing business as holonic systems and implementing holonic system algorithms and by viewing business as self-organising organisational structures realising self-organising business models like the ones mentioned in Sect. 2.1.3. Examples of each category are given below.

2.2.10.1 Complex Adaptive WfMSs

Complex adaptive WfMSs are increasingly used to support adaptive business processes with typical examples being dynamic workflow management [27] and intelligent manufacturing scheduling [85]. They are based on distributed software components executing without central top-down control and possibly in an asynchronous manner. Each component may be designed with different goals and structure such that the resulting behaviour is not practically deducible from a specification of its parts in a formal way. Such systems can exhibit emergent behaviour with unpredictable results. For example, an emergent phenomenon in complex adaptive WfMSs supporting supply chain management is that the variation in the demand experienced by a low-tier supplier is much wider than that experienced by the OEM, sometimes increasing by as much as two times from one tier to the next [79].

In a different approach, complex adaptive WfMSs can be developed in terms of autonomous agents interacting in groups according to local rules. The global emergent behaviour results from local interactions between agents. In such systems, nonlinearity is considered as the main cause of emergent behaviour [57, 87]. There are various sources for nonlinearity in business systems. For example, in manufacturing systems three main sources of nonlinearity are capacity limits, feedback loops and temporal delays [79].

2.2.10.2 Holonic WfMSs

The holonic WfMSs aim to support *holonic enterprises* [70]. This concept has emerged from the need for flexible open, reconfigurable models, able to emulate the market dynamics in the networked economy. Due to market dynamics, it is required that business strategies and relationships evolve over time and are modified according to the dynamic business environment.

The main idea in the holonic enterprise model stems from the work of Arthur Koestler [60]. Koestler postulated a set of underlying principles to explain the self-organising tendencies of social and biological systems. Koestler has identified structural patterns of self-replicating structures in natural systems, which he named *holarchies*, and he proposed the term *holon* to describe the elements of these systems. Holarchies have been envisioned as models of the Universe's self-organising structure, which consist of holons, at several levels of resolution in the nested hierarchy [28]. Holons behave as autonomous wholes, and yet, as cooperative parts for achieving the goal of the holarchy, and therefore holons can be regarded as nested agents. In such a nested hierarchy, each holon is a sub-system retaining the characteristic attributes of the whole system.

A holonic enterprise is a holarchy of collaborative enterprise units or even individual enterprises, where each enterprise unit is regarded as a holon. Holons represent both physical and logical entities such as production departments and machines. The holonic enterprises paradigm provides a framework for information and resource management in global virtual organisations by modelling enterprise entities as software agents linked through the Internet [104]. In this parallel universe of information, enterprises enabled with the above mechanism can evolve towards better and better structures while at the same time self-organising their resources to optimally accomplish the desired objectives. Holonic WfMSs implement the holonic enterprise model considering that holonic structures are dynamically changing through self-organisation. Self-organisation is achieved using appropriate holon configuration meta-models, such as centrally optimised or supervisory configuration and dynamic reconfiguration based on experimental data [94].

To determine the optimum holonic structure, various evolutionary and genetic approaches can be used. A representative example is the fuzzy-evolutionary approach proposed by Ulieru [91]. That approach clusters the entities of a holonic enterprise into a dynamically configured optimal holonic structure and applies self-organisation and evolution models inspired from natural systems. Self-organisation is induced by *minimising the entropy*. Entropy minimisation is achieved by measuring the information spread across the system and acting towards reaching an equilibrium involving optimal interaction between the system's parts.

The evolution of the current system into a better one is achieved by interaction with external systems which are located via *genetic search strategies*. These search strategies mimick mating with most fit partners in natural evolution, and they inject new information in the system. The new optimal organisational structure of the system, reached by minimising the entropy in self-organisation, is then better than the one before evolution. Latest applications of the above approach have been reported in the design of Internet-enabled soft computing holarchies for telemedicine, for example e-Health, telehealth and telematics [92]. More details about self-organisation of holonic structures are provided in Chap. 11.

2.2.10.3 Self-organising Workflow Models

Self-organising business models are frequently used to implement self-organisation in WfMSs. Such models are implemented inherently in the software infrastructure. For example, Helal and Wang [47] propose an agent-based infrastructure and appropriate interaction protocols that would be useful in negotiating service bundles for the different services offered by agents. In that approach, agents are organised in e-business communities (Malls). Within a community, highly relevant agents group together offering special e-Services for a more effective, mutually beneficial and more opportune e-Business. Self-organisation is based on a hierarchy of brokers which control membership and operations of the different communities. The aim of the broker hierarchy is to achieve scalability of the system and interoperability of the various heterogeneous e-business components.

Furthermore, in many cases pre-existing business and social models are modified to use the benefits of self-organisation. For example, interest by IBM, as part of their *Autonomic Computing* [58] program, and by Microsoft, as part of the *Dynamic Systems Initiative* [72], indicates the importance of self-organisation for managing distributed resources and shaping the way business is carried out in dynamic environments. In addition, applications based on invisible intelligent technology have started being embedded in clothes, walls and cars, enabling users to carry out completely new operations such as personalised virtual shopping, e-purse-based micropayments and context-based traffic guidance [35]. The pervasive, large-scale and embedded nature of these applications leads naturally to design them based on self-organisation so that to handle the vital issues of robustness and adaptability among others.

2.3 A FAQ About Self-organisation and Emergence

This section attempts to summarise answers to fundamental questions about self-organising systems and to provide our view on the subject. The question–answer

Table 2.1 A FAQ about self-organising systems

Question	Answer
What is Self-organisation?	Self-organisation is the process with which a system changes its structure without any external control to respond to changes in its operating conditions and its environment.
What is Emergence?	Emergence is the phenomenon where a non-predetermined outcome, such as a structure or a state, is reached progressively following multiple self-organisation acts of the system.
When is a system Self-organising?	A system is self-organising when it is able to apply self-organisation and emergence in order to function as needed in a dynamic environment.
What are the *Self-** properties?	The Self-* properties are system capabilities which can be exercised without any external control. They are all covered by the more generic term 'self-organisation'. Examples of self-* properties are self-management, self-healing and self-adaptation.
What are examples of natural systems showing self-organising behaviour?	Typical examples of self-organising systems are the human immune system, ant and bee swarms, schools of fish and flocks of birds.
What are examples of applications of self-organising software?	Typical examples of self-organising software applications include adaptive manufacturing control, P2P systems, network coordination and security, collective robotics and workflow management.
What are the main benefits of self-organisation?	The main benefits of self-organisation are increased robustness, scalability and individual unit communication and processing costs. Furthermore, self-organisation enables the engineering of underspecified software.
What are the main drawbacks of self-organisation?	The main drawbacks of self-organisation include the possibility of occurrences of opposing actions among system units, the appearance of needless redundancies in unit behaviours and the lack of guarantees for finding the globally optimum solution.

format used is a very effective way to provide a succinct introduction to the subject of self-organising software. See Table 2.1 above.

2.4 Conclusion

Self-organising systems have initially been observed in nature, one of the most cited example being the termites behaviour following a stigmergy mechanism reported by Grassé. Their appealing characteristics is the relative simple behaviour of the individual system' components when compared to the complex collective behaviour they produce. As artificial software systems become more difficult to manage and maintain, researchers and industrials have started to take inspiration from those systems and to translate their underlying mechanisms into artificial systems. Many applications domains have been considered ranging from static and dynamic optimisation problems (resource allocation, manufacturing control) based on ant foraging behaviour, to network security following basic immune systems principles, to workflow management for the holonic enterprise. This chapter provided a short introduction to several natural and artificial systems highlighting their specificities.

2.5 Problems–Exercises

2.1 Given the initial understanding of the self-organisation and emergence concepts obtained after reading this chapter:

(a) Provide three examples of natural systems where self-organisation and emergence are inherently applied.
(b) For the provided examples, describe how self-organisation is carried out. For example, in the case of ant foraging self-organisation is carried out using stigmergy.
(c) For the provided examples, describe what are the emergent outcomes or states where applicable. For example, in the case of foraging ants an emergent outcome is the path between nest and food source which is formed by foraging ants.

2.2 Consider a system consisting of an automatic door and a sensor. The door opens automatically when the sensor identifies some movement within a specific distance and radius. If there is no movement, the door remains in the closed position or returns to it if it was open.

(a) Do you consider this system as self-organising? Justify your answer.
(b) Identify the emergent outcomes that can possibly result from system operation, if any.

2.3 Provide three examples of application areas where self-organising software is used. For each one:

(a) Discuss the benefits gained from using self-organisation as compared to using a traditional system.
(b) Identify conditions where the use of a deterministic, centralised system would be more appropriate.

Key Points

- Introduction to well-known natural self-organising systems, such as ant foraging, chemical reactions, immune systems or business systems;
- Introduction to self-organising application domains, such as, among others, Computational Grids, network security, swarm robotics or sensor networks;
- An FAQ table for a quick review of introductory notions.

2.6 Further Reading

The Origins of Order: *Self-organization and Selection in Evolution*. A foundational introduction to self-organisation and emergence in complex systems. (S. Kaufman, 1993, Oxford University Press.)
Swarm Intelligence: *From Natural to Artificial Systems*. A comprehensive book on self-organising algorithms drawn from natural systems. (E. Bonabeau, M. Dorigo and G. Theraulaz, 1999, Oxford University Press.)
Self-organization in Biological Systems. A detailed presentation of self-organisation mechanisms in biological systems. (S. Camazine, J.-L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz and E. Bonabeau, 2001, Princeton University Press.)
Fundamentals of Natural Computing: *Basic Concepts, Algorithms, and Applications*. A comprehensive introduction, including appropriate algorithms, to evolutionary computing, swarm intelligence and immune systems. (Leandro Nunes de Castro, 2006, Chapman and Hall.)
The many Facets of Natural Computing. An up-to-date review article on recent advances in natural computing. (L. Kari and G. Rosenberg, 2008, Communications of the ACM, 51(10):72–83.)
Self-managed Systems and Services. An overview of the uses of self-organisation in implemented systems and services. (J.P. Martin-Flatin, J. Sventek and K. Geihs, 2006, Guest Editorial of special issue, Communications of the ACM, 49(3): 36–39.)

That special issue includes additional relevant papers as well.

Emergent computation: *self-organizing, collective, and cooperative phenomena in natural and artificial computing network*. An overview of emergence in computing systems. (Stephanie Forrest, 1991, Special issue of Physica D, MIT Press.)

More foundational readings are provided at the end of Chap. 3.

References

1. http://www.entropia.com (2003)

2. http://www.gnutella.com (2003)
3. http://www.kazaa.com (2003)
4. http://www.ud.com (2003)
5. http://www.swarm-bots.org/ (2005)
6. http://www.swarmanoid.org/ (2010)
7. http://www.symbrion.eu/ (2013)
8. Aberer, K., Datta, A., Hauswirth, M.: P-Grid: dynamics of self-organizing processes in structured peer-to-peer systems. In: Peer-to-Peer Systems and Applications. LNCS, vol. 3485, pp. 137–153. Springer, Berlin (2005)
9. Appleby, S., Steward, S.: Mobile software agents for control in telecommunications networks. BT Technol. J. **12**(2), 104–113 (1994)
10. Babaoglu, O., Meling, H., Montresor, A.: Anthill: a framework for the development of agent-based peer-to-peer systems. In: 22th International Conference on Distributed Computing Systems (ICDCS'02), pp. 15–22. IEEE Computer Society, Los Alamitos (2002)
11. Bar-Yam, Y.: Dynamics of Complex Systems. Perseus Books, Cambridge (1997)
12. Barabási, A., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
13. Bartholdi, J.J., Eisenstein, D.D.: A production line that balances itself. Oper. Res. **44**(1), 21–34 (1996)
14. Bartholdi, J.J., Eisenstein, D.D.: Using bucket brigades to migrate from craft manufacturing to assembly lines. Manuf. Serv. Oper. Manag. **7**(2), 121–129 (2005)
15. Bartholdi, J.J., Eisenstein, D.D., Foley, R.D.: Performance of bucket brigades when work is stochastic. Oper. Res. **49**(5), 710–719 (2001)
16. Bonabeau, E., Henaux, F., Guerin, S., Snyers, D., Kuntz, P.: Routing in telecommunications networks with "smart" ant-like agents. In: Intelligent Agents for Telecommunications Applications '98 (IATA'98) (1998)
17. Bonabeau, E., Dorigo, M., Théraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, London (1999)
18. Bonabeau, E., Dorigo, M., Theraulaz, G.: Inspiration for optimisation from social insect behaviour. Nature **406**, 39–42 (2000)
19. Brooks, R.: Artificial life: from robot dreams to reality. Nature **406**, 945–947 (2000)
20. Brueckner, S., Parunak, H.V.D.: Self-organising manet management. In: Serugendo, G., Karageorgos, A., Rana, O.F., Zambonelli, F. (eds.) Engineering Self-Organising Systems, Nature-Inspired Approaches to Software Engineering. Lecture Notes in Artificial Intelligence, vol. 2977, pp. 20–35. Springer, Berlin (2004)
21. Buhler, P.A., Vidal, J.M.: Towards adaptive workflow enactment using multiagent systems. Inf. Technol. Manag. **6**(1), 61–87 (2005). doi:10.1007/s10799-004-7775-2
22. Bussmann, S., Jennings, N.R., Wooldridge, M.: Multiagent Systems for Manufacturing Control. Springer, Berlin (2004)
23. Buyurgan, N., Meyyappan, L., Saygin, C., Dagli, C.H.: Real-time routing selection for automated guided vehicles in a flexible manufacturing system. Int. J. Manuf. Technol. Manag. **18**(2), 169–181 (2007)
24. Camazine, S., Deneubourg, J.L., Franks, N.R., Sneyd, J., Théraulaz, G., Bonabeau, E.: Self-Organization in Biological Systems. Princeton University Press, Princeton (2001). 2nd edn. (2003)
25. Cardelli, L.: Machines of systems biology. In: Bulletin of the EATCS 93, pp. 176–204 (2007)
26. Castro, L.D., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer, Berlin (2002)
27. Chen, Q., Hsu, M., Dayal, U., Griss, M.: Multi-agent cooperation, dynamic workflow and XML for e-commerce automation. In: Sierra, C., Gini, M., Rosenschein, J.S. (eds.) Proceedings of the Fourth International Conference on Autonomous Agents, Barcelona, Catalonia, Spain, pp. 255–263. ACM, New York (2000)
28. Christensen, J.H.: Holonic manufacturing systems: initial architecture and standards directions. In: Proceedings of the First European Conference on Holonic Manufacturing Systems (1994)

29. Clair, G., Kaddoum, E., Gleizes, M.P., Picard, G.: Self-regulation in self-organising multi-agent systems for adaptive and intelligent manufacturing control. In: Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO'08, 20–24 October 2008, pp. 107–116. IEEE Computer Society, Los Alamitos (2008)
30. Craig, W.R.: Flocks, herds and schools: a distributed behavioral model. SIGGRAPH Comput. Graph. **21**(4), 25–34 (1987)
31. Deneubourg, J.L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., Chretien, L.: The dynamics of collective sorting robot-like ants and ant-like robots—simulation of animal behaviour. In: Meyer, J., Wilson, S. (eds.) First International Conference on Simulation of Adaptive Behaviour, pp. 356–363. MIT Press, Cambridge (1991)
32. Di Caro, G., Dorigo, M.: Ant colonies for adaptive routing in packet switched communication networks. In: Proceedings of PPSN V—Fifth International Conference on Parallel Problem Solving from Nature. Lecture Notes in Computer Science. Springer, Berlin (1998)
33. Di Marzo Serugendo, G., Karageorgos, A., Rana, O.F., Zambonelli, F. (eds.): Engineering Self-Organising Systems: Nature-Inspired Approaches to Software Engineering. Lecture Notes in Computer Science, vol. 2977. Springer, Berlin (2004)
34. Di Marzo Serugendo, G., Gleizes, M.P., Karageorgos, A.: Self-organisation in multi-agent systems. Knowl. Eng. Rev. **20**(2), 165–189 (2005)
35. Ducatel, K., Bogdanowicz, M., Scapolo, F., Leitjen, J., Burgelman, J.C.: Scenarios for ambient intelligence in 2010. Technical Report, Institute for Prospective Technological Studies (2001)
36. Endy, D.: Foundations for engineering biology. Nature **438**, 449–453 (2005)
37. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On power-law relationships of the Internet topology. In: Proceedings of the Special Interest Group on Data Communications (ACM SIGCOMM'99), pp. 251–262. ACM, New York (1999)
38. Fernandez-Marquez, J.L., Arcos, J.L., Di Marzo Serugendo, G.: A decentralized approach for detecting dynamically changing diffuse event sources in noisy WSN environments. In: International Conference on Self-Adaptive and Self-Organising Systems (SASO'10). IEEE Computer Society, Los Alamitos (2010)
39. Flake, G.W., Lawrence, S., Giles, C.L., Coetzee, F.M.: Self-organisation and identification of web communities. Computer **35**(3), 66–71 (2002)
40. Forrest, S., Hofmeyer, S.A., Somayaji, A., Longstaff, T.A.: A sense of self for Unix processes. In: Proceedings of the IEEE Symposium on Research in Security and Privacy, pp. 120–128. IEEE Computer Society, Los Alamitos (1996)
41. Foukia, N.: IDReAM: intrusion detection and response executed with agent mobility. In: International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'05), Utrecht, The Netherlands, pp. 264–270. IEEE Press, New York (2005)
42. Georgé, J.P., Peyruqueou, S., Régis, C., Glize, P.: Experiencing self-adaptive MAS for real-time decision support systems. In: Demazeau, Y., Pavón, J., Corchado, J., Bajo, J. (eds.) 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009). Advances in Intelligent and Soft Computing, vol. 55, pp. 302–309. Springer, Berlin (2009)
43. Grassé, P.P.: La reconstruction du nid et les coordinations interindividuelles chezbellicositermes natalensis etcubitermes sp la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. Insectes Soc. **6**(1), 41–80 (1959). doi:10.1007/BF02223791
44. Gross, R., Dorigo, M.: Towards group transport by swarms of robots. Int. J. Bio-Inspired Comput. **1**(1/2), 1–13 (2009)
45. Haken, H.: Synergetics: an introduction. Springer, Berlin (1978), reedited (1983)
46. Hardaker, G., Graham, G.: Energizing your e-commerce through self-organising collaborative marketing networks. Tech. Rep. (2002)
47. Helal, S., Wang, M.: Service centric brokering in dynamic e-business agent communities. J. Electron. Commer. Res. **2**(1), 32–47 (2001). Special Issue on Intelligent Agents in E-commerce

48. Heylighen, F.: The science of self-organization and adaptivity. In: Kiel, L.D. (ed.) Knowledge Management, Organizational Intelligence and Learning, and Complexity, The Encyclopedia of Life Support Systems, EOLSS, pp. 253–280. Eolss Publishers, Oxford (2001)
49. Holland, O., Melhuis, C.: Stigmergy, self-organisation and sorting in collective robotics. Artif. Life **5**(2), 173–202 (1999)
50. Horst, F.W., Muddassar, F.: A comprehensive review of nature inspired routing algorithms for fixed telecommunication networks. J. Syst. Archit. **52**(8), 461–484 (2006)
51. Hulaas, J., Binder, W., Di Marzo Serugendo, G.: Enhancing Java grid computing security with resource control. In: International Conference on Grid Services Engineering and Management (GSEM'04) (2004)
52. Itao, T., Suda, T., Aoyama, T.: Jack-in-the-net: adaptive networking architecture for service emergence. In: Asian-Pacific Conference on Communications (2001)
53. Jelasity, M., Babaoglu, O.: T-man: Gossip-based overlay topology management. In: Engineering Self-Organizing Systems. LNCS, vol. 3910, pp. 1–15. Springer, Berlin (2006)
54. Jennings, N.R., Wooldridge, M.J. (eds.): Agent Technology: Foundations, Applications and Markets. Springer, Berlin (1998). This is a cross-referenced BOOK (collection) entry
55. Kari, L., Rosenberg, G.: The many facets of natural computing. Commun. ACM **51**(10), 72–83 (2008)
56. Karuna, H., Valckenaers, P., Saint-Germain, B., Verstraete, P., Zamfirescu, C.B., Van Brussek, H.: Emergent forecasting using stigmergy approach in manufacturing coordination and control. In: Engineering Self-Organizing Applications (ESOA'04), pp. 210–226. Springer, Berlin (2004)
57. Kaufman, S.: At Home in the Universe: The Search for the Laws of Self-Organization and Complexity. Oxford University Press, London (1995)
58. Kephart, J.O., Chess, D.M.: The vision of autonomic computing. Computer **36**(1), 41–50 (2003). doi:10.1109/MC.2003.1160055
59. Ko, S.Y., Gupta, I., Jo, Y.: Novel mathematics-inspired algorithms for self-adaptive peer-to-peer computing. In: International Conference on Self-Adaptive and Self-Organising Systems (SASO'07) (2007)
60. Koestler, A.: The Ghost in the Machine. Hutchison and Co, London (1967), the Danube edn.
61. Levi, P., Kernbach, S. (eds.): Symbiotic Multi-Robot Organisms. Cognitive Systems Monographs, vol. 7. Springer, Berlin (2010)
62. Li, Y., Chen, F., Sun, X., et al.: Self-adaptive resource management for large-scale shared clusters. J. Comput. Sci. Technol. **25**(5), 945–957 (2010)
63. Luhmann, N.: Familiarity, confidence, trust: problems and alternatives. In: Gambetta, D. (ed.) Trust: Making and Breaking Cooperative Relations, pp. 213–237 (2000). Chap. 13. Published Online http://www.sociology.ox.ac.uk/papers/gambetta213-237.pdf
64. Lynden, S., Rana, O.F.: Coordinated learning to support resource management in computational grids. In: Second IEEE Conference on Peer-to-Peer Computing (P2P 2002), pp. 81–89. IEEE Computer Society, Los Alamitos (2002)
65. Mamei, M., Vasirani, M., Zambonelli, F.: Self-organizing spatial shapes in mobile particles: the TOTA approach. In: Brueckner, S., Di Marzo Serugendo, G., Karageorgos, A., Nagpal, R. (eds.) Engineering Self-Organising Systems. Lecture Notes in Computer Science, vol. 3464, pp. 138–153. Springer, Berlin (2005)
66. Martin-Flatin, J., Sventek, J., Geihs, K.: Self-managed systems and services. Commun. ACM **49**(3), 36–39 (2006)
67. Martino, S.: A mobile agent approach to intrusion detection. Tech. Rep., Joint Research Center—Institute for Systems, Informatics and Safety (1999)
68. Maturana, H.R., Varela, F.J.: Autopoiesis and Cognition: The Realization of the Living. Boston Studies in the Philosophy of Science. Springer, Berlin (1991). 2nd edn. (1980), 1st edn. (1973)
69. Mazhar, N., Farooq, M.: BeeAIS: artificial immune system security for nature inspired, MANET routing protocol, BeeAdHoc. In: Artificial Immune Systems. LNCS, vol. 4628, pp. 370–381. Springer, Berlin (2007)

70. McHugh, P., Merli, G., Wheeler, W.A.: Beyond Business Process Reengineering: Towards the Holonic Enterprise. Wiley, New York (1995)
71. Meyyappan, L., Soylemezoglu, A., Saygin, C., Dagli, C.H.: A wasp-based control model for real-time routing of parts in a flexible manufacturing system. Int. J. Comput. Integr. Manuf. **21**(3), 259–268 (2008)
72. Microsoft: Dynamic systems 2007: Get started with dynamic systems technology today. Tech. Rep. (2007)
73. Mills, K.: A brief survey of self-organization in wireless sensor networks. Wirel. Commun. Mob. Comput. **7**, 823–834 (2007)
74. Montresor, A., Meling, H., Babaoglu, O.: Messor: load-balancing through a swarm of autonomous agents. In: Moro, G., Koubarakis, M. (eds.) Agents and Peer-to-Peer Computing. Lecture Notes in Artificial Intelligence, vol. 2530, pp. 125–137. Springer, Berlin (2003)
75. Nedelec, F., Surrey, T., Karsenti, E.: Self-organisation and forces in the microtubule cytoskeleton. Curr. Opin. Cell Biol. **15**(2), 118–124 (2003)
76. Nicolis, G., Prigogine, I.: Self-Organization in Non-Equilibrium Systems. Wiley, New York (1977)
77. O'Hare, G.M.P., Jennings, N.R. (eds.): Foundations of Distributed Artificial Intelligence. Software Agents. Wiley, London (1996)
78. Parunak, H.V.D.: Synthetic pheromone mechanisms for coordination of unmanned vehicles. In: AAMAS, Bologna, Italy (2002)
79. Parunak, H.V.D., VanderBok, R.S.: Managing emergent behavior in distributed control systems. In: ISA-Tech'97, Anaheim (1997)
80. Picard, G., Gleizes, M.P.: The adelfe methodology: designing adaptive cooperative multiagent systems. In: Bergenti, F., Gleizes, M.P., Zambonelli, F. (eds.) Methodologies and Software Engineering for Agent Systems. The Agent-Oriented Software Engineering Handbook, vol. 11. Kluwer Academic, Amsterdam (2004)
81. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall Series. Prentice Hall, New York (1995)
82. Schoonderwoerd, R., Holland, O., Bruten, J.: Ant-like agents for load balancing in telecommunications networks. In: 1st International Conference on Autonomous Agents, pp. 209–216. IEEE Computer Society, Los Alamitos (1997)
83. Searle, J.R.: The Rediscovery of the Mind. MIT Press, Cambridge (1992)
84. Seeley, T.D.: When is self-organization used in biological systems? Biol. Bull. **202**, 314–318 (2002)
85. Shen, W., Norrie, D.H.: Agent-based systems for intelligent manufacturing: a state-of-the-art survey. Knowl. Inf. Syst. **1**(2), 129–156 (1999)
86. Smartdust project. http://robotics.eecs.berkeley.edu/~pister (2003)
87. Stewart, M.: The Coevolving Organization. Decomplexity Associates LtD, Rutland (2001)
88. Thakoor, S., Morookian, J.M., Chahl, J., Hine, B., Zornetzer, S.: Bees: exploring mars with bioinspired technologies. Computer **37**(9), 36–47 (2004)
89. Ueda, K.: Emergent synthesis approaches to biological manufacturing systems. In: Cunha, P.F., Maropoulos, P.G. (eds.) Digital Enterprise Technology: Perspectives and Future Challenges, pp. 25–34. Springer, New York (2007)
90. Ueda, K., Markus, A., Monostori, L., Kals, H.J.J., Arai, T.: Emergent synthesis methodologies for manufacturing. CIRP Ann. Manuf. Technol. **50**(2), 535–551 (2001). doi:10.1016/S0007-8506(07)62994-1
91. Ulieru, M.: Emergence of holonic enterprises from multi-agent systems: a fuzzy evolutionary approach. In: Soft Computing Agents, vol. 13, pp. 187–215. IOS Press, Amsterdam (2002)
92. Ulieru, M.: Internet-enabled soft computing holarchies for e-health applications (soft computing enhancing the Internet and the Internet enhancing soft computing). In: Zadeh, L.A., Nikravesh, M. (eds.) New Directions in Enhancing the Power of the Internet. Springer, Berlin (2003)
93. Ulieru, M.: Adaptive information infrastructures for the e-society. In: Brueckner, S., Serugendo, G.D.M., Karageorgos, A., Nagpal, R. (eds.) Engineering Self-Organising Systems:

Methodologies and Applications. Lecture Notes in Artificial Intelligence, vol. 3464, pp. 32–51. Springer, Berlin (2005)

94. Valckenaers, P., Brussel, H.V.: Holonic manufacturing execution systems. CIRP Ann. Manuf. Technol. **54**(1), 427–432 (2005)
95. Valckenaers, P., Brussel, H.V., Kollingbaum, M., Bochmann, O.: Multi-agent coordination and control using stigmergy applied in manufacturing control. In: Luck, M., Marik, V., Stepankova, O., Trappl, R. (eds.) Multi-Agent Systems and Applications, 9th ECCAI Advance Cource ACAI 2001 and AgentLink's 3rd European Agent Systems Summer School, EASS 2001, Selected Tutorial Papers, vol. 2086, pp. 317–334. Springer, Berlin (2001)
96. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: Prosa. Comput. Ind. **37**(3), 255–274 (1998). doi:10.1016/S0166-3615(98)00102-X
97. Visual behaviours for mobile robots project. http://www.isr.ist.utl.pt/vislab/projects.html (2003)
98. Viroli, M., Zambonelli, F.: A biochemical approach to adaptive service ecosystems. Inf. Sci. **180**(10), 1876–1892 (2010)
99. Visconti, A., Tahayori, H.: Detecting misbehaving nodes in MANET with an artificial immune system based on type-2 fuzzy sets. In: IEEE Proceedings of the 4th International Conference for Internet Technology and Secured Transactions (ICITST09). IEEE Computer Society, Los Alamitos (2009)
100. Vittikh, V.A., Skobelev, P.O.: Multi-agent systems for modelling of self-organization and cooperation processes. In: XIII International Conference on the Application of Artificial Intelligence in Engineering, Galway, Ireland, pp. 91–96 (2002)
101. Werbach, K.: Syndication—the emerging model for business in the Internet era. Harvard Bus. Rev. **78**, 85–93 (2000)
102. Wokoma, I., Sacks, L., Marshall, I.: Biologically inspired models for sensor network design. In: London Communications Symposium (2002)
103. Wrzesinska, G., Maassen, J., Bal, H.: Self-adaptive applications on the Grid. In: 12th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (2007)
104. Zhang, X., Norrie, D.H.: Holonic control at the production and controller levels. In: Proceedings of the 2nd International Workshop on Intelligent Manufacturing Systems, pp. 215–224 (1999)