Self-Organisation: Paradigms and Applications

Giovanna Di Marzo Serugendo, Noria Foukia, Salima Hassas,
Anthony Karageorgos, Soraya Kouadri Mostéfaoui,
Omer F. Rana, Mihaela Ulieru, Paul Valckenaers, Chris Van Aart

Engineering Self-Organising Applications Working Group
http://www.agentcities.org/Activities/WG/ESOA/

Abstract. A self-organising system functions without central control, and through contextual local interactions. Components achieve a simple task individually, but a complex collective behaviour emerges from their mutual interactions. Such a system modifies its structure and functionality to adapt to changes to requirements and to the environment based on previous experience. Nature provides examples of self-organisation, such as ants food foraging, molecules formation, or antibodies detection. Similarly, current software applications are driven by social interactions (negotiations, transactions), based on autonomous entities or agents, and run in highly dynamic environments. The issue of engineering applications, based on the principles of self-organisation to achieve robustness and adaptability, is gaining increasing interest in the software research community. The aim of this paper is to survey natural and artificial complex systems exhibiting emergent behaviour, and to outline the mechanisms enabling such behaviours.
Keywords: Self-organisation, self-organising application, emergence, collective behaviour, multi-agent systems.

1 Introduction

The study of self-organising systems is a field that has been explored at least since 1953 with the work done by Grassé [25], who studied the behaviour of insect societies. Many systems in nature demonstrate self-organisation, such as planets, cells, organisms and societies. All these systems exhibit recurrent properties inherent to self-organisation. The simplest form of self-organisation can be achieved by the arrangement of parts of a system in such a way as to be non-random. Considerable research has already been undertaken to study such systems. Self-organising systems are often encountered in many scientific areas including biology, chemistry, geology, sociology, and information technology.

A large number of software self-organising systems are designed based on natural mechanisms of self-organisation. Furthermore, recent research has been oriented towards introducing self-organisation mechanisms specifically for software applications, as well as entire development techniques supporting self-organisation [19]. 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.

This survey firstly defines self-organisation, and presents examples of self-organising systems taken from natural life. Subsequently, it describes the different mechanisms enabling social, natural and artificial organisations to achieve a coherent global behaviour. Finally, it reviews several software applications exhibiting a self-organising behaviour.

2 Self-Organising Systems

The notion of self-organisation is popular in many different research fields. Therefore, it is difficult to find a precise and concise definition of what is the meaning of the term self-organisation. However, it seems that similar properties are apparent and recurrent among the different definitions and research fields referring to self-organisation. The next sub-section defines what are the inherent characteristics of self-organisation. Subsequently, the following sub-sections describe a number of different types of self-organising systems. It is not the intention of this paper to provide an exhaustive review of all types of self-organising systems. Instead, self-organising systems are classified in three broad categories: physical, living and social systems. For each category a representative example is described.

2.1 Definition

Intuitively, self-organisation refers to the fact that a system's structure or organisation appears without explicit control or constraints from *outside* the system. In other words, the organisation is intrinsic to the self-organising system and results from *internal* constraints or mechanisms, due to *local interactions* between its components [11]. These interactions are often *indirect* thanks to the environment [25]. The system dynamics modifies also its *environment*, and the modifications of the external environment influence in turn the system, but without disturbing the internal mechanisms leading to organisation. The system evolves *dynamically* [9] either in time or space, it can maintain a stable form or can show transient phenomena. In fact, from these interactions, *emergent* properties appear transcending the properties of all the individual sub-units of the system [30]. One well-known example is that of a colony of ants sorting eggs without having a particular ant knowing and applying the sorting algorithm. The emergence is the fact that a structure, not explicitly represented at a lower level, appears at a higher level. With *no central control*, a complex collective behaviour raises then from simple local individual interactions. More generally, the field of complex systems studies emergent phenomenon, and self-organisation [8].

2.2 Physical Systems

A characteristic of physical self-organising systems is that many of them present a so called *critical value* in which the state of the system changes suddenly to another state under certain conditions (temperature, pressure, speed, ...). Thus, self-organisation is observed *globally* when the physical system moves from a chaotic disordered state to a stable one. For instance, a thermodynamic system such as a gas of particles has emergent properties, temperature and pressure, that do not derive from the description of an individual particle, defined by its position and velocity. Similarly, chemical reactions create new molecules that have properties that none of the atoms exhibit before the reaction takes place [8]. Moreover, the magnetisation of a multitude of spins[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 thanks to a strong emerging magnetic field.

2.3 Living Systems

A scientific aspect in the study of living organisms is the determination of invariant in the evolution of natural systems. In particular the spontaneous appearance of an order in living complex systems due to the self-organisation. In the optic of biological research, the global emergence of a behaviour or a feature that can not be reduced to the properties of each system's component (molecules, agents, cells, ...) defines also the common meaning of self-organisation. One example described in [43] is the self-organisation of the cytoskeleton thanks 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 of the higher organisms, containing a true nucleus bounded by a nuclear membrane. These cells are founded in animals and plants. Eukaryotic cells are often organised into organs to create higher levels of complexity and function thanks to metabolic processes. The obtained organ has a defined functionality that transcends all the functionality offered by each of its constitutive cells. These cells are the basic functioning units of life and evolve in step through external changes with the environment. These units of life (cells) have to use internal metabolic processes such as mutation or natural selection, to adapt to natural life's evolution. This is also the result of the self-organisation of cells. Other examples in human body are the human nervous system, or the immune system. Such living systems transparently manage vital functions, such as blood pressure, digestion, or antibodies creation.

2.4 Social Systems

Social insects organise themselves to perform activities such as food foraging or nests building. Cooperation among insects is realised through an indirect

[1] A spin is a tiny magnet.

communication mechanism, called stigmergy, and by interacting through their environment. Insects, such as ants, termites, or bees, mark their environment using a chemical volatile substance, called the pheromone, for example, as do ants to mark a food trail. Insects have simple behaviour, and none of them alone "knows" how to find food but their interaction gives rise to an organised society able to explore their environment, find food and efficiently inform the rest of the colony. The pheromonal information deposited by insects constitutes an indirect communication means through their environment.

Apart from animal societies, human beings organise themselves into advanced societies. Human beings use direct communication, they engage in negotiation, build whole economies and organise stock markets.

3 Self-Organising Mechanisms

This section presents the major self-organising mechanisms used in natural and software systems to achieve self-organisation.

3.1 Magnetic Fields

A self-organisation phenomenon has been studied in the structure of a piece of potentially magnetic material. A magnetic material consists of a multitude of tiny magnets or spins. The spins point in different directions cancelling their respective magnetic fields. At a lower level, the orientation of the spins is due to the random movements of the molecules in the material: the higher the temperature, the stronger these random movements of the molecules in the material. These molecular movements affect the spins making them difficult to orient in an ordered way. However, if the temperature decreases the spins spontaneously point in the same direction. In this case, the different magnetic fields now add up, producing a strong overall field. Magnetisation exhibits self-organisation because the orientation of the spins is variable and depends on the local neighbourhood. Under low temperature, the force between neighbouring spins is dominating and they tend to build order. Similar phenomena, are observed in the crystallisation from a liquid state, which is another common example of self-organisation.

3.2 Kohonen Neural Networks

Kohonen neural networks, also called self-organising maps, are useful for clustering applications [36]. They take their inspiration from brain cells, which are activated depending on the subject's location. Such a network is made of two neurons layers (input, and output), and usually follows a regular two-dimensional grid of neurons. This grid represents a topological model of the application to cluster. Indeed, the network maps similar data to the same, or adjacent, node of the grid, by projecting multi-dimensional vectors of data onto a two-dimensional regular grid, preserving the data clustering structure.

To each neuron is associated a weight vector, randomly initialised. This topology preserving behaviour of the network is obtained through a learning rule which determines the *winner* among the neurons, as being the one whose weight vector is closer to the vector of the sampled data entered in the network. Data will then be affected to this neuron. Once the winner is determined, an update of the weight vectors of each neuron is performed, in order to reinforce clustering. On the basis of this algorithm, a method, called WEBSOM [37], has been defined, which helps organise heterogeneous text documents onto significant maps.

3.3 Stigmergy

Social insect societies (ants, bees, wasps, termites, etc) exhibit many interesting complex behaviours, such as emergent properties from local interactions between elementary behaviours achieved at an individual level. The emergent collective behaviour is the outcome of a process of self-organisation, in which insects are engaged through their repeated actions and interactions with their evolving environment [30]. Self-organisation in social insects relies on an underlying mechanism, the mechanism of stigmergy, first introduced by Grassé in 1959 [25]. Grassé studied the behaviour of a kind of termites during the construction of their nests and noticed that the behaviour of workers during the construction process is influenced by the structure of the constructions themselves. This mechanism is a powerful principle of cooperation in insect societies. It has been observed within many insect societies like those of wasps, bees and ants. It is based on the use of the environment as a medium of inscription of past behaviours effects, to influence the future ones. This mechanism defines what is called a self-catalytic process, that is the more a process occurs, the more it has chance to occur in the future. More generally, this mechanism shows how simple systems can produce a wide range of more complex coordinated behaviours, simply by exploiting the influence of the environment. Much behaviour in social insects, such as foraging or collective clustering are rooted on the stigmergy mechanism. Foraging is the collective behaviour through which ants collect food by exploring their environment. During the foraging process, ants leave their nest and explore their environment following a random path. When an ant finds a source of food, it carries a piece of it and returns back to the nest, by laying a trail of a hormone called pheromone along its route. This chemical substance persists in the environment for a particular amount of time before it evaporates. When other ants encounter a trail of pheromone, while exploring the environment, they are influenced to follow the trail until the food source, and enforce in their coming back to the nest the initial trail by depositing additional amounts of pheromone. The more a trail is followed, the more it is enforced and has a chance of being followed by other ants in the future. Collective Sorting is a collective behaviour through which some social insects sort eggs, larvae and cocoons [17]. As mentioned in [10], an ordering phenomenon is observed in some species of ants when bodies of dead ants are spread in the foraging environment. Ants pick up dead bodies and drop them later in some area. The probability of picking up an item is correlated with the density of items in the region where the operation occurs. This behaviour has been studied in robotics through simulations [29] and real implementations [30]. Robots with primitive behaviour are able to achieve a spatial environment structuring, by forming clusters of similar objects via the mechanism of stigmergy described above.

Tag-Based Models. In addition to the digital pheromone, which is the artificial counterpart of the natural pheromone used by the ants, new electronic mechanisms directly adapted to software applications are being developed. The notion of tags, a mechanism from simulation models, is one of them. Tags are markings attached to each entity composing the self-organising application [26]. These markings comprise certain information on the entity, for example functionality and behaviour, and are observed by the other entities. In this case the interaction would occur on the basis of the observed tag. This would be particularly useful to let interact electronic mobile devices that do not know each other in advance. Whenever they enter the same space, for example a space where they can detect each other and observe the tags, they can decide on whether they can or cannot interact.

3.4 Coordination

The social aspect of multi-agent systems is engineered through coordination models that define the agent interactions in terms of interaction protocols and interaction rules. In other words, a coordination model defines how agents interact and how their interactions can be controlled [15]. This includes dynamic creation and destruction of agents, control of communication flows among agents, control of spatial distribution and mobility of agents, as well as synchronisation and distribution of actions over time [12]. In general, a coordination model is defined by: (a) *coordinable entities* (components), these are the agents, which are coordinated. Ideally, these are the building blocks of a coordination architecture, for example agents, processes, tuples, atoms, etc.; (b) *coordination media*, these are the coordinators of inter-agent entities. They also serve to aggregate a set of agents to form a configuration, for example channels, shared variables, tuple spaces; and (c) *coordination laws* ruling actions by coordinable entities or the coordination media. The laws usually define the semantics of a number of coordination mechanisms that can be added to a host language [16].

A particular coordination model depends on the coordination media, coordination laws, and the programming language used for expressing coordinables. In control-driven models agents interact either by broadcasting events to the other agents, or through a point-to-point channel connection. Communication among the agents is established by a third party coordinator process. Coordination laws establish propagation of events, dynamic reconnection of ports, and creation and activation of processes.

In data-driven models, the coordination media is a shared data space, and interactions consists in asynchronously exchanging data through the data space.

Coordination laws govern data format and primitives for storing, removing, and retrieving them from the interaction space. Such models are derived from Linda [24], an early coordination model based on shared tuple spaces. This model fits well with multi-agent systems because it allows interactions among anonymous entities, joining and leaving the system continuously, in a not-predefined manner. In order to incorporate more control over interactions, into data-driven models, hybrid models are also considered. In such models, the data space serves for responding to communication events. It becomes a programmable coordination medium where new coordination primitives and new behaviours can be added in response to communication events. Acting as middleware layers, coordination spaces provide uncoupled interaction mechanisms among autonomous entities, which input data into a common tuple space, and may retrieve data provided by other entities. These models support a limited form of self-organisation, since they enable decentralised control, anonymous and indirect local interactions among agents.

Tuples On The Air. On top of the basic coordination environment, several enhancements have been realised in order to address specifically self-organisation. The TOTA environment (Tuples On The Air) propagates tuples, according to a propagation rule, expressing the scope of propagation, and possible content change [39]. If we come back to the metaphor of ant societies, such a model allows, among others, to electronically capture the notion of digital pheromone, deposited in the tuple space and retrieved by other agents. The propagation rule removes the pheromone from the data space, once the evaporation time has elapsed.

Coordination Fields. Alternatively, the Co-Fields (coordination fields) model drives agents behaviour as would do abstract force fields [40]. The environment is represented by fields, which vehicle coordination information. Agents and their environment create and spread such fields in the environment. A field is a data structure composed of a value (magnitude of field), and a propagation rule. An agent then moves by following the coordination field, which is the combination of all fields perceived by the agent. The environment updates the field according to the moves of the agents. These moves modify the fields which in turn modify the agents behaviour. This model allows representing not only complex movements of ants, and birds, but also tasks division and succession.

4 Self-Organising Applications

Nature provides examples of emergence and self-organisation. Current distributed applications, as well as applications of a near future, already show a self-organising behaviour, since they are situated in highly changing environments, they are made of a large number of heterogeneous components and cannot undergo a central control.

4.1 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 to 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 [46, 32].

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 [44]. 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 usually follow a coordination model as explained in Subsection 3.4. 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, interaction are not pre-defined, and there is no global system goal. The interaction dynamics between an agent and its environment lead to emergent structure or emergent functionality, even though no component is responsible for producing a global goal.

4.2 Grid

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). Computational engines can comprise of specialist, tightly coupled architectures (such as parallel machines) or loosely coupled clusters of workstations. There has been an emerging interest in trying to integrate resources across organisational boundaries through file or CPU sharing software (such as KaZaA [4] and Gnutella [3] for file sharing and Entropia [2] and UD [5] for CPU sharing). These individual resources are often geographically distributed, and may be owned by different administrators (or exist within different independently administered domains). Managing resources within Computational Grids is currently based on infrastructure with centralised registry and information services (based on the LDAP/X500 directory service) - such as provided by the Open Grid Services Infrastructure (OGSI) [1]. In this process, resource owners must register their capabilities with a limited number of index servers, enabling subsequent search on these servers by resource users. The provision of such centralised servers is clearly very limiting, and restricts the scalability of such approaches. Current resources being provided within Computational Grids are owned by national or regional centres (or by research institutions), and therefore concerns regarding access rights and usage need to be pre-defined and approved. However, as resources from less trusted users are provided, the need to organise these into dynamic communities, based on a number of different criteria: performance, trust, cost of ownership and usage, usability, etc. become significant. Self-organisation therefore plays an important role in identifying how such communities may be formed and subsequently dis-banded. A utility-based approach for forming such communities is explored in [38]. However, it is necessary to understand and investigate alternative incentive structures that will enable the formation of such communities.

4.3 Service Emergence.

Itao et al. [31] 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, or 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's 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.

4.4 Web Communities

Flake et al. [21] have shown using a specific algorithm that Web pages form related communities. A Web page is a member of a community if it has more hyperlinks within the community than outside it. At the human level, one cannot have an overall picture of the structure, based on hyperlinks, that emerges among the Web pages. 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 simply 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. Indeed, it is now possible to reach existing pages from a new location and it is 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.

4.5 Networking with Ants

Di Caro et al. [18] suggests using artificial ant-based modelling to solve network problems. The motivation is that ants modelling might be able to cope with communication networks better than humans. A first survey [9], dealing with several swarm intelligence examples in social insect communities, shows how ants-like behaviour (ants, bees, termites, and wasps) provide a powerful metaphor to build a completely decentralised system. Such a system is composed of individual and simple entities, which collaborate to allow a more complex and collective behaviour. The global emergent behaviour of the ant population is due to a network of interactions between the ants themselves, but also between the ants and their environment. This emergent collective behaviour allows the social insect colony to organise vital tasks such as finding food, building the nest, dividing labour among individuals and spreading alarm among members of the society. Many of these tasks and their respective mechanism have inspired computer network scientists notably to mimic ant foraging behaviour to optimise the routing in communication networks or to mimic the division of labour and the task allocation to optimise the load balancing in network systems. When ants forage, they wander randomly starting from their source (nest) until they reach their destination (food) and on their way they lay a chemical trail termed *pheromone*. The pheromone deposited along the path followed by each ant marks this path for other ants. In fact, the more one path is marked the more it will be chosen by other ants. This mechanism where the environment becomes the communication medium is termed *stigmergy* (see Subsection 3.3). To apply this paradigm to network routing, Dorigo and his colleagues built an artificial ant network where periodically each node launches an ant to find the route to a given destination.

By simply smelling the strength of the pheromones along the neighbourhood paths of the node, the ant generates the map that shows the fastest route to any end point. In case of congestion, it was showed that this mechanism outperforms all other popular routing algorithms in terms of speed achieved to avoid the traffic jams.

4.6 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 [41]. 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 [23]. 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 [22]. 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.

4.7 Robots

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 [7] 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.

4.8 Manufacturing Control

The food foraging behaviour in ant colonies has been translated into a design for agent societies performing manufacturing control. Resource agents provide a reflection of the underlying production system in the world of the agents. These resource agents offer a space for the other agents to navigate through - each agent knowing its neighbours - and offer spaces on which information can be put, observed and modified - like the ants leave pheromones in the physical world. Virtual agents - 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, ants explore possible routings for the products being made, make a selection and propagate the resulting intentions through the 'reflection'. 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) [51].

4.9 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 [54].

Sensor networks benefit also of 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 [6].

4.10 Business Process Infrastructures

Business Process Infrastructures (BPIs) are software infrastructures 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 BPIs need to be able to self-organise, namely adapt their functionality to support changes in business process requirements. The majority of BPIs are based on the agent metaphor. Agents provide flexibility and adaptability and therefore they are particularly suitable for realising self-organising infrastructures. Self-Organising BPIs are characterised by three main features: The underlying model which is used to conceptualise the business operations, the software paradigm used to develop the necessary software components and the method used to engineer the self-organising and emergent properties. Based on the underlying business model, self-organising BPIs can be classified as complex interactive BPIs, holonic BPIs, and organisational BPIs.

Complex interactive BPIs. Complex interactive BPIs are increasingly used to support business processes, for example in dynamic workflow management [13] and intelligent manufacturing scheduling [47]. 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 interactive BPIs 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 [45]. Complex Interactive BPIs 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, non-linearity is considered as the main cause of emergent behaviour [33, 48]. There are various sources for non-linearity in business systems. For example in manufacturing systems three main sources of nonlinearity are capacity limits, feedback loops, and temporal delays [45].

Holonic BPIs. The Holonic BPIs aim to support *holonic enterprises* [42]. This concept has emerged from the need for flexible open, reconfigurable models, able to emulate the market dynamics in the networked economy, which necessitates that strategies and relationships evolve over time, changing with the dynamic business environment.

The main idea of the holonic enterprise model stems from the work of Arthur Koestler [35]. Koestler postulated a set of underlying principles to explain the self-organising tendencies of social and biological systems. Starting from the empirical observation that, from the Solar System to the Atom, the Universe is organised into self-replicating structures of nested hierarchies, intrinsically embedded in the functionality of natural systems, Koestler has identified structural patterns of self-replicating structures named *holarchies*. Koestler proposed the term holon to describe the elements of these systems, which is a combination of the Greek word *holos*, meaning "whole", with the suffix *on* meaning "part" as in proton or neuron [49]. Holarchies have been envisioned as models for the Universe's self-organising structure, in which holons, at several levels of resolution in the nested hierarchy [14], behave as autonomous wholes, and yet, as cooperative parts for achieving the goal of the holarchy. As such, 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. What actually defines a holarchy is a purpose around which holons are clustered and subdivided in sub-holons, at several levels of resolution, according to the organisational level of dissection required.

A holonic enterprise is a holarchy of collaborative enterprises, where each enterprise is regarded as a holon and is modelled by a software agent with holonic properties, so that the software agent may be composed of other agents that behave in a similar way, but perform different functions at lower levels of resolution. A holon represents, as well, an autonomous and co-operative entity of a holonic enterprise, which includes operational features, skills and knowledge, and individual goals. Holons represent both physical and logical entities such as production departments and machines. A holon has information about itself and the environment, containing an information processing part, and often a physical processing part. An important feature of holonic enterprises is that a holon can be part of another holon, for example, a holon can be broken into several others holons, which in turn can be broken into further holons, which allows the reduction of the problem complexity.

Self-organisation of the system is achieved by using appropriate configuration meta-models of various types, such as centrally optimised or supervisory configuration and dynamic reconfiguration based on experimental data. To determine the optimum holonic structure various evolutionary and genetic approaches can be used. For example, the fuzzy-evolutionary approach, proposed by Ulieru in [49], clusters the entities of a holonic enterprise into a dynamically configured optimal holonic structure. It mimics self-organisation and evolution of natural systems as follows. On one side, self-organisation is induced by *minimising the entropy*, measuring the information spread across the system, such that equilibrium involving optimal interaction between the system's parts is reached. On the other side, it enables system's evolution into a better one by enabling interaction with external systems found via *genetic search strategies* (mimicking mating with most fit partners in natural evolution), such that the new system's optimal organisational structure (reached by minimising the entropy) is better than the one before evolution.

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 [55]. 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. Encapsulating the dynamic evolutionary search strategy into a mediator agent and designing the virtual clustering mechanism by the fuzzy entropy minimisation strategy above, empowers the holonic enterprises with self-adapting properties. Moreover, the holonic enterprise evolves like a social organism in Cyberspace, by mating its components with new partners as they are discovered in a continuous, incremental improvement search process. Latest applications have shown high suitability for this strategy to the design of Internet-enabled soft computing holarchies for telemedicine, for example e-Health, telehealth and telematics [50].

Organisational BPIs. Organisational BPIs aim to support business in the context of global economy. The difference between Complex Interactive BPIs, Holonic BPIs and Organisational BPIs is that the latter view businesses from a global economy perspective and are focused on business theoretic models of self-organisation, such as marketing, management, and economic models.

For example, one such model from the marketing domain is the one-to-one variable pricing model [27] 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 [53]. The company allows customers to access computer systems, via the Web site, to monitor the status of their packages. For corporate customers FedEx provide software tools that enable the 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 an Organisational BPI model from the area of management is the theory of activity described in [52]. The view of a company is that it consists of networks of working groups that can change their structure, links and behaviour in response to business requirements. The aim is to capture the self-organisation decisions that need to be taken during the business operations both by managers and by interactions between employees with emphasis to solving potential conflicts of interests both of the inner and outside co-operative activity of the company.

In other cases the focus is on the technological part of the infrastructure. For example Helal and Wang in [28] 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.

5 Conclusion

There is currently a growing interest in biologically inspired systems, not only from researchers but also from industry. Recent interest by IBM, as part of their *Autonomic Computing* [34] program, and by Microsoft, as part of the *Dynamic Systems Initiative*, indicates the importance of self-organisation for managing distributed resources. In addition, future applications based on invisible intelligent technology will be made available in clothes, walls, or cars, and people can freely use it for virtual shopping, micro-payment using e-purses, or traffic guidance system [20]. These applications, by their pervasive, large-scale, and embedded nature exhibit self-organisation characteristics, and would gain in robustness, and adaptability if considered and programmed as self-organising systems.

This survey shows that most current techniques used for designing self-organising applications are direct translations of natural self-organising mechanisms, such as: the immune system, brain cells, magnetic fields, or the stigmergy paradigm. Translating natural mechanisms into software applications is a first step. However, there is a need to define specific mechanisms for self-organisation that fit electronic applications. Recent examples in that direction are given by the TOTA infrastructure (subsection 3.4), or those that can be built on the notion of tags (subsection 3.3). The challenges to take up in this field go beyond interaction mechanisms and middleware technologies favouring self-organisation. They relate to the establishment of whole software engineering methodologies encompassing design, test, and verification, based on these mechanisms as well as on sound mathematical theories enabling the definition of local goals, given the expected global behaviour.

6 Acknowledgments

The research described in this paper is partly supported by the EU funded Agentcities.NET project (IST-2000-28384). The opinions expressed in this paper are those of the authors and are not necessarily those of the EU Agentcities.NET partners. Giovanna Di Marzo Serugendo is supported by Swiss NSF grant 21-68026.02.

References

1. http://forge.gridforum.org/projects/ogsi-wg.
2. http://www.entropia.com/.
3. http://www.gnutella.com/.
4. http://www.kazaa.com/.
5. http://www.ud.com/.
6. SmartDust project, http://robotics.eecs.berkeley.edu/~pister/SmartDust/.
7. Visual behaviors for mobile robots project, http://www.isr.ist.utl.pt/vislab/projects.html.
8. Y. Bar-Yam. *Dynamics of Complex Systems.* Perseus Books, Cambridge, MA, 1997.
9. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* Santa Fe Institute Studies on the Sciences of Complexity. Oxford University Press, UK, 1999.
10. E. Bonabeau, V. Fourcassié, and J.-L. Deneubourg. The phase-ordering kinetics of cemetery organization in ants. Technical Report 98-01-008, Santa Fe Institute, 1998.
11. S. Camazine, J.-L. Deneubourg, Nigel R. F., J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organization in Biological System.* Princeton Studies in Complexity. Princeton University Press, 2001.
12. N. Carriero and D. Gelernter. Tuple Analysis and Partial Evaluation Strategies in the Linda Compiler. In *Second Workshop on Languages and Compilers for Parallel Computing*, pages 115–125. MIT Press, 1989.
13. Q. Chen, M. Hsu, U. Daya, and M. Griss. Multi-Agent Cooperation, Dynamic Workflow and XML for E-Commerce Automation. In C. Sierra, M. Gini, and J. Rosenschein, editors, *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 255–263. ACM Press, 2000.
14. J.H. Christensen. Holonic manufacturing systems: Initial architecture and standards directions. In *Proceedings of the First European Conference on Holonic Manufacturing Systems*, 1994.
15. P. Ciancarini. Multiagent coordination: A computer science perspective. In Y. Demazeau, editor, *Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'01)*, 2001.
16. P. Ciancarini, A. Omicini, and F. Zambonelli. Multiagent system engineering: The coordination viewpoint. In N. R. Jennings and Y. Lespérance, editors, *Intelligent Agents VI. Agent Theories, Architectures, and Languages*, volume 1757 of *LNAI*, pages 250–259. Springer-Verlag, 2000.
17. J.-L. Deneubourg, S. Goss, N. Francks, A. Sendova-Francks, C. Detrain, and L. Chretien. The dynamic of collective sorting: Robot-like ants and and-like robots. In J. A. Meyer and S. W. Wilson, editors, *Proceedings of the First International Conference on Simulation and Adaptive Behavior: From Animals to Animat*, pages 356–363. MIT Press, 1991.
18. G. Di Caro and M. Dorigo. Ant colonies for adaptive routing in packet-switched communications networks. In *Proceedings of PPSN V - Fifth International Conference on Parallel Problem Solving from Nature*, volume 1498 of *LNCS*. Springer-Verlag, 1998.
19. G. Di Marzo Serugendo, A. Karageorgos, O. F. Rana, and F. Zambonelli, editors. *First International Workshop on Engineering Self-Organising Applications (ESOA'03)*, 2003.

20. K. Ducatel, M. Bogdanowicz, F. Scapolo, J. Leijten, and J.-C. Burgelman. Scenarios for Ambient Intelligence in 2010. Technical report, Institute for Prospective Technological Studies, 2001.
21. G. W. Flake, S. Lawrence, C. L. Giles, and F. M. Coetzee. Self-organization and identification of web communities. *IEEE Computer*, 35(3):66–71, 2002.
22. S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff. A sense of self for Unix processes. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pages 120–128. IEEE Computer Society Press, 1996.
23. N. Foukia, S. Hassas, S. Fenet, and J. Hulaas. An intrusion response scheme: Tracking the source using the stigmergy paradigm. In *Proceedings of the Security of Mobile Multi-Agent Systems Workshop (SEMAS'02)*, July 2002.
24. D. Gelernter. Generative communication in linda. *ACM Transactions on Programming Languages and Systems*, 7(1):80–112, January 1985.
25. P. P. Grassé. La reconstruction du nid et les interactions inter-individuelles chez les bellicositermes natalenis et cubitermes sp. la théorie de la stigmergie: essai d'interprétation des termites constructeurs. *Insectes Sociaux*, 6:41–83, 1959.
26. D. Hales and B. Edmonds. Evolving Social Rationality for MAS using "Tags". In J. S. Rosenschein, T. Sandholm, M. Wooldridge, and M. Yokoo, editors, *Second International Joint Conference on Autonomous Agents and MultiAgent Systems*, pages 495–503. ACM Press, 2003.
27. G. Hardaker and G. Graham. *Energizing your e-Commerche through Self.* Salford, 2002.
28. S. Helal, M. Wang, and A. Jagatheesan. Service-centric brokering in dynamic e-business agent communities. *Journal of Electronic Commerce Research (JECR), Special Issue in Intelligent Agents in E-Commerce*, 2(1), February 2001.
29. O. Holland. Multi-agent systems: Lessons from social insects and collective robotics. In Sandip Sen, editor, *Working Notes for the AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems*, pages 57–62, March 1996.
30. O. Holland and C. Melhuis. Stimergy, self-organization, and sorting in collective robotics. *Artificial Life*, 5(2):173–202, 1999.
31. T. Itao, T. Suda, and T. Aoyama. Jack-in-the-net: Adaptive networking architecture for service emergence. In *Proceedings of the Asian-Pacific Conference on Communications*, 2001.
32. N. Jennings, K. Sycara, and M. Wooldridge. A Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, 1(1):7–38, 1998.
33. S. Kauffman. *At Home in the Universe - The Search for the Laws of Self-Organization and Complexity.* Oxford University Press, 1997.
34. J. O. Kephart and D. M. Chess. The Vision of Autonomic Computing. *Computer*, 36(1):41–50, January 2003.
35. A. Koestler. *The Ghost in the Machine.* Arkana, 1967.
36. T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences.* Springer, 3rd edition edition, 2001.
37. K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. WEBSOM for textual data mining. *Artificial Intelligence Review*, 15(5/6):345–364, December 1999.
38. S. Lynden and O. F. Rana. Coordinated learning to support resource management in computational grids. In *Proceedings of 2nd IEEE Conference on Peer-to-Peer Computing (P2P 2002)*, pages 81–89. IEEE Computer Society, 2002.
39. M. Mamei and F. Zambonelli. Self-Organization in MultiAgent Systems: a Middleware approach. In G. Di Marzo Serugendo, A. Karageorgos, O. F. Rana, and

F. Zambonelli, editors, *First Workshop on Engineering Self-Organising Applications (ESOA'03)*, 2003.

40. M. Mamei, F. Zambonelli, and L. Leonardi. Co-fields: Towards a unifying approach to the engineering of swarm intelligent systems. In *3rd International Workshop on Engineering Societies in the Agents World (ESAW)*, number 2577 in LNCS, pages 68–81. Springer-Verlag, 2003.
41. S. Martino. A mobile agent approach to intrusion detection. Technical report, Joint Research Center - Institute for Systems, Informatics and Safety, June 1999.
42. P. McHugh, G. Merli, and W. A. Wheeler. *Beyond Business Process Reengineering: Towards the Holonic Enterprise.* John Wiley and Sons, 1995.
43. F. Nedelec, T. Surrey, and E. Karsenti. Self-organisation and forces in the microtubule cytoskeleton. *Current Opinion in Cell Biology*, 15(2):118–124, 2003.
44. G. O'Hare and N. R. Jennings. *Foundations of Distributed Artificial Intelligence.* John Wiley and Sons, New York, USA, 1996.
45. H. V. D. Parunak and R. S. VanderBok. Managing emergent behavior in distributed control systems. In *Proceedings of ISA Tech '97, Instrument Society of America*, 1997.
46. S. Russel and P. Norvig. *Artificial Intelligence: a Modern Approach.* Prentice-Hall, 1995.
47. W. Shen and D. H. Norrie. Agent-based systems for intelligent manufacturing: a state-of-the-art survey. *Knowledge and Information Systems, an International Journal*, 1(2):129–156, May 1999.
48. M. Stewart. *The Coevolving Organization.* Decomplexity Associates LtD, 2001.
49. M. Ulieru. Emergence of holonic enterprises from multi-agent systems: A fuzzy-evolutionary approach. In V. Loia, editor, *Soft Computing Agents: A New Perspective on Dynamic Information Systems*, pages 187–215. IOS Press, 2002.
50. M. Ulieru. Internet-enabled soft computing holarchies for e-health applications. In L. A. Zadeh and M. Nikravesh, editors, *New Directions in Enhancing the Power of the Internet.* Springer-Verlag, 2003. To appear.
51. P. Valckenaers, H. Van Brussel, M. Kollingbaum, and O. Bochmann. Multi-agent coordination and control using stigmergy applied to manufacturing control. In M. Luck, V. Mařik, O. Stepánková, and R. Trappl, editors, *Multi-Agent Systems and Applications, 9th ECCAI Advanced Course ACAI 2001 and Agent Link's 3rd European Agent Systems Summer School, EASSS 2001, Selected Tutorial Papers*, volume 2086 of *LNCS*, pages 317–334. Springer-Verlag, 2001.
52. V. A. Vittikh and P. O. Skobelev. Multi-agents systems for modelling of self-organization and cooperation processes. In *XIII International Conference on the Application of Artificial Intelligence in Engineering*, pages 91–96, 1998.
53. K. Werbach. Syndication - the emerging model for business in the internet era. *Harvard Business Review*, pages 85–93, May-June 2000.
54. I. Wokoma, L. Sacks, and I. Marshall. Biologically inspired models for sensor network design. In *London Communications Symposium*, 2002.
55. H. Zhang and D. H. Norrie. Holonic control at the production and controller levels. In *Proceedings of the 2nd International Workshop on Intelligent Manufacturing Systems*, pages 215–224, 1999.